RSA NETWITNESS® SUITE

Alerting Using ESA Guide

for Version 11.0

Contact Information

RSA Link at https://community.rsa.com contains a knowledgebase that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

Trademarks

For a list of RSA trademarks, go to www.emc.com/legal/emc-corporation-trademarks.htm#rsa.

License Agreement

This software and the associated documentation are proprietary and confidential to EMC, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by EMC.

Third-Party Licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed on the product documentation page on RSA Link. By using this product, a user of this product agrees to be fully bound by terms of the license agreements.

Note on Encryption Technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

October 2017

Contents

Getting Started with ESA

This topic covers quick start topics for RSA NetWitness® Suite Event Stream Analysis (ESA) to help you get started in using ESA. The following topics are designed to assist you in working with ESA Correlation Rules.

- Best Practices helps you to understand how to best set up, deploy, and create rules.
- Troubleshoot ESA helps you to troubleshoot different aspects of ESA, including rule writing and deployment.
- View Memory Metrics for Rules helps you to work with memory metrics to understand memory usage for ESA services.

There are two ESA services that can run on an ESA host:

- Event Stream Analysis (ESA Correlation rules)
- Event Stream Analytics Server (ESA Analytics)

The first service is the Event Stream Analysis service that creates alerts from ESA rules, also known as ESA Correlation Rules, which you create manually or download from Live. This user guide covers alerting using ESA Correlation Rules. For information on configuring ESA Correlation Rules, see the "Configure ESA Correlation Rules" section of the *ESA Configuration Guide*.

The second service is the ESA Analytics service, which is used for Automated Threat Detection. Because the ESA Analytics service uses preconfigured ESA Analytics modules for Automated Threat Detection, you do not have to create or download rules to use it. For information on the ESA Analytics service, see the *Automated Threat Detection Guide* and the "Configure ESA Analytics" section of the *ESA Configuration Guide*.

Best Practices

Best practices provide guidelines to help you write and manage rules, deploy rules, and maintain system health for your ESA services.

Understand Event Stream Analysis Rule Types

The Event Stream Analysis service provides advanced stream analytics such as correlation and complex event processing at high throughputs and low latency. It is capable of processing large volumes of disparate event data from Concentrators. However, when working with Event Stream Analysis, you should be aware of the factors that affect resource usage in order to create effective rules.

Each event that is received by ESA is evaluated to determine if it may trigger a rule. There are three types of rules that can be deployed in order to determine what the ESA engine should do with the incoming event. Each of these rule types have different impacts on system resource utilization. All three rule types may be created via the Rule Builder, Advanced EPL rules, or downloaded via RSA Live. The table below lists the rule type and the impact this rule may have on system resources.

Rule Type	Description
Simple Filter Rule	This rule has no correlation to other events. At ingestion time, this rule is evaluated against a set of conditions, and if those conditions are met an alert is generated. If no conditions match, the event is quickly released by the engine to free up memory usage. These rules do not take up memory since the events are not retained beyond the initial evaluation. The memory resource usage does not increase as more simple filter rules are deployed. However, if the filter condition is too generic, it is possible that this rule can generate too many alerts, which will strain the system resources for the storage and retrieval of these alerts. For example, you might write a rule to generate an alert when HTTP network activity arrives over a non-standard HTTP port.
Event Window Rule	This rule evaluates a set of events over a time period for specific conditions. At ingestion time, the rule is evaluated against a set of conditions. If those conditions are met, the event is retained in memory for a specific amount of time. After the specified time passes, the events are removed from the time window if the number of events collected does not meet the threshold to trigger an alert. The memory consumption of such rules are highly dependent on the incoming event rate (traffic), the amount of data per event, and the time length specified in the event window. Each matching event is retained in memory until the time window has passed, so the longer the time window, the greater the potential volume. For example, you might write a rule that generates an alert if a user fails to log into any system five times within a ten minute time frame.

Rule Type	Description
Followed By Rule	This rule evaluates a chain of incoming events to determine if the sequence of events matches a particular condition. At ingestion time, the rule is evaluated against a set of conditions. If the conditions are met, one of two actions occurs: • If this is the first event of the sequence, a new event thread is started, and the event is retained as the head of the sequence. • If the event belongs to an existing event thread, it is added to that sequence. In both cases, the event is retained in memory. The amount of resource usage is particularly sensitive to the customer environment for this type of rule. If the filter condition generates many event threads, resources are consumed for each new thread (in addition to the event). Additionally, if the end of the event thread is never met (i.e., an alert is never generated), then the entire event is saved in memory indefinitely. For example, you might write a rule to generate an alert when a user fails to log in to a server, then performs a successful login, and then creates a new account.

In addition to the memory usage discussed above, alert generation also consumes system resources. Each alert that is generated must be stored for retrieval and must also be processed by NetWitness Respond. This process uses disk space for storage, requires database memory to be consumed, and increases CPU utilization running queries.

When writing and deploying rules, you should be aware that each of these actions "cost" you system resources. The sections below are designed to help you keep your usage at a healthy level and monitor for problems if systems are becoming overloaded.

Best Practices for Writing Rules

These are general guidelines for writing rules.

- **Create alerts for actionable events.** The purpose of an alert should be to notify you of an event that requires immediate and specific action. For events that do not require action, or only require you to have awareness of the event, you can create a report.
- **Configure new rules as trial rules so you can observe how they react in your environment**. If you deploy new rules as trial rules, they will be disabled if the configured memory threshold is exceeded. You can also use the memory snapshot feature to see how much memory was being used when a trial rule was disabled. For more details, see Work with Trial Rules.

- **Configure Alert notifications only after your rule testing and tuning is complete**. This can help ensure you do not get flooded with notifications if a rule behaves differently than you expect.
- **Rules need to be specific so that you limit resource usage**. Use the following guidelines to limit usage:
  - Make the filters on the rule exclude all but the necessary events for the rule to fire accurately.
  - Make the size of your windows (window time for correlation) as small as possible.
  - Limit the events that you include in the window: For example, if you only want to see IDS events, ensure that you only include those events in your time window.
- **Rules need to be tuned to an alert level that is manageable**. If you are flooded with alerts, then the purpose and utility of an alert is lost. For example, maybe you want to know about encrypted traffic to other countries. But, you could limit the list to countries that are known risks. This limits the volume of alerts to a level you can manage.

Best Practices for Working with RSA Live Rules

These are guidelines for RSA Live Rules.

- **Deploy RSA Live rules in small batches**. Not every rule is suited to every environment. The best way to ensure your RSA Live rules are successful is to deploy them in small batches so you can test them in your environment. If you deploy small batches, it's much easier to tell if a particular rule has an issue.
- **Read the rule descriptions provided with RSA Live rules**. ESA rules are not “one size fits all.” Not all rules will work in your environment. The rule descriptions tell you which parameters you will need to modify to successfully deploy a rule in your environment.
- **Set your parameters**. RSA Live rules have parameters that need to be modified. If you do not modify your parameters, the rule may not work or it may exhaust your memory.
- **Deploy new rules as trial rules so you can observe how they react in your environment.** If you deploy new rules as trial rules, they will be disabled if the configured memory threshold is exceeded. For more details, see Work with Trial Rules.

Best Practices for Deploying Rules

These are general guidelines for deploying rules.

- **Deploy rules in small batches so you can observe how they react in your environment.** Not all environments are the same, and a rule will need to be tuned for memory usage, alert volume, and effective detection of events.
- **Test rules before you configure alert notifications**. Configure Alert notifications only after your rule testing and tuning is complete. This can help ensure you do not get flooded with alerts if a rule behaves differently than you expect.
- **Monitor system health as a part of your deployment process**. When you deploy rules, monitor your system's health as a part of your deployment process. You can view total memory utilization for your ESA in the Health and Wellness tab. For more information, see "Viewing Health and Wellness statistics" in Troubleshoot ESA.

Best Practices for System Health

These are general guidelines for system health.

- **Set up new rules as trial rules**. A common issue is that new rules may cause memory issues. To prevent this, you can set up new rules as trial rules. If the configured memory threshold is met, all trial rules are disabled to prevent the system from running out of memory. For more information about trial rules, see Work with Trial Rules.
- **Set up thresholds in the Health & Wellness module to alert you if memory usage is too high**. There are metrics in the Health & Wellness module that track memory usage. You can set up alerts and notifications to send you an email if those thresholds are crossed. For more information about the memory statistics you can view, see "Viewing Health and Wellness statistics" in Troubleshoot ESA.
- **Monitor memory metrics for each rule in the Health & Wellness module**. For each rule, you can view the estimated memory usage in the Health & Wellness module. You can use this information to ensure that rules do not use too much memory. For more information about the memory statistics you can view, see "Viewing Health and Wellness statistics" in Troubleshoot ESA.

Troubleshoot ESA

This section describes common issues that may occur while using ESA, and it suggests common solutions to these problems.

Troubleshoot ESA Services

Problem	Possible Causes	Solutions
On the NetWitness Suite Dashboard, the ESA service appears in red to indicate it is offline. In the **CONFIGURE** > **ESA Rules** view, the following message appears: "The Service is either offline or not reachable."	Several	When an ESA service is offline, there are many possible causes. However, a common issue is that you have created a rule that uses excessive memory and causes the ESA service to fail. To troubleshoot this problem, see Steps to Troubleshoot Memory Issues with an ESA Service Offline. Other common causes might be that your firewall is blocking the connection between the ESA and NetWitness Suite, or the ESA service machine may be down.
		To bring up ESA Services: From **ADMIN** > **Services**, select the actions icon  for your ESA service, and choose **start**. If your ESA service is stopping and restarting in a loop, you may need to call Customer Support to get the services to start.
After a recent upgrade, the ESA service appears in red on the NetWitness Suite Dashboard to indicate it is offline. In the **CONFIGURE** > **ESA Rules** view, the following message appears: "The Service is either offline or not reachable."	Configuration issues	If your system has been recently upgraded, you may have made a configuration error. Under **ADMIN** > **Services**, select your ESA service, and click **Edit Service**. On the Edit Service field, click Test Connection. If the connections fails, you likely have a configuration error. Attempt to fix your configuration error, and try again.

Problem	Possible Causes	Solutions
The ESA appears to be running slowly.	Configuration issues	You may be able to improve performance by modifying the buffer (the default value is *1048576 bytes*), or setting the TCP setting to TCPNoDelay to prevent a delay in receiving TPC acknowledgments (Acks). You can modify these settings (*readBufferSize* and *tcpNoDelay*) by going to */Workflow/Source/nextgenAggregation* in the Explore view.

Troubleshoot RSA Live Rules for ESA

Problem	Possible Causes	Solutions
I imported a group of rules from RSA Live, and now my ESA service is crashing. Why?	You may not have configured the parameters for the RSA Live rule to tune it for your environment.	Each rule in RSA Live has a description that includes the parameters you must configure and prerequisites for your environment. Review this description to see if the rule is appropriate for your environment. To ensure that you deploy rules safely in your environment, configure new rules as trial rules to test them in your environment. Trial rules add a safeguard for testing new rules. For details on this, see Deploy Rules as Trial Rules.

Problem	Possible Causes	Solutions
I imported a group of rules from RSA Live, and while the rules deployed without errors, they were later disabled.	Not all RSA Live rules are meant for every environment. You may not have the correct meta in your ESA for the rule to run.	You can verify that a rule was disabled by going to **CONFIGURE > ESA Rules > Services > Deployed Rule Stats**. If the rule is disabled, the green icon does not display next to the rule. If a rule deployed correctly but was disabled, check the logs for exceptions related to the rule. Specifically, check to see if the rules were disabled due to missing meta. To do this, go to **ADMIN** > **Services**, select your ESA service and then > **View** > **Logs**. Then, search for a message similar to the following: "Property named '<meta_name>' is not valid in any stream" For example, you might see: `Failed to validate filter expression '(medium=1 and streams=2 or medium=3...(238 chars)': Property named 'tcp_flags_seen' is not valid in any stream` If a similar message displays, you may need to add a custom meta key to the Log Decoder or Concentrator. To do this, follow these instructions: "Create Custom Meta Keys Using Custom Feed " in the *Decoder and Log Decoder Configuration Guide*.

Troubleshoot Deployments

Problem	Possible Causes	Solutions
I created a rule, and I checked the syntax. The rule looked fine. When I went to deploy the rule, I got an error. Why?	You may not have the correct meta to deploy the rule.	Check the Meta key references. You may not have the correct meta to deploy the rule.

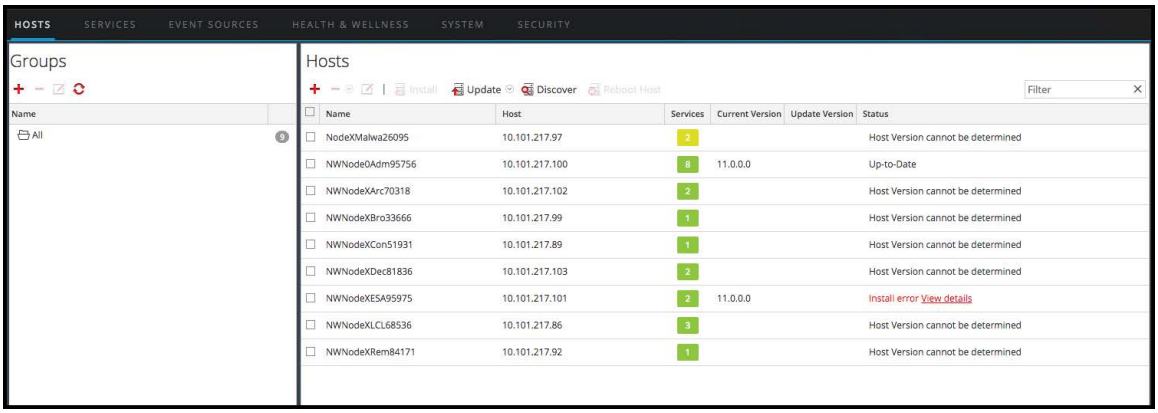
Troubleshoot Rules

Problem	Possible Causes	Solutions
I created a custom rule (via the Rule Builder or Advanced EPL), and my rule is not firing. Why?	You may have connectivity issues.	Check the 'Offered Rate' statistic on the **CONFIGURE** > **ESA Rules** > **Services** tab. If the offered rate is zero, then the ESA service is not receiving data from Concentrators. Validate the Concentrator connectivity. Go to **ADMIN** > **Services**, select your ESA, and then **View** > **Config**. Ensure the concentrator is enabled. Select the concentrator and click on **test connection**. If the offered rate is not zero, the meta key name and type used in the rule likely doesn't match the meta key present in events. Check to see if the meta key name and type used in the rule is valid by searching for the meta key name in **CONFIGURE** > **ESA Rules** > **Settings** tab (Meta key references search).
	There may be a problem with the rule.	**If a specific rule is not firing**, go to **CONFIGURE** > **ESA Rules** > **Services** to see if the rule was disabled. In the **Deployed Rule Stats** section, a rule that is disabled displays a clear enabled button (instead of the green enabled button). You can also check Events Matched field. Go to **CONFIGURE** > **ESA Rules** > **Services**. From there, you can see the number of events that were matched in the **Events Matched** column. If no events matched, check the logic of your rule for errors. For example, check the syntax for uppercase and lowercase errors, and check the time window. If the rule still doesn't fire, consider simplifying the logic of the rule to see if it fires when there is less complexity.

Steps to Troubleshoot Memory Issues with an ESA Service Offline

Step 1: Verify that your Host Is Running

The first step to troubleshooting is to ensure that your host is running. To do this, go to **ADMIN**> **HOSTS**. If the host is down, the system parameters will not display (updating host information can sometimes be delayed), the **Services** display in red, and the **Updates** field displays an error message.



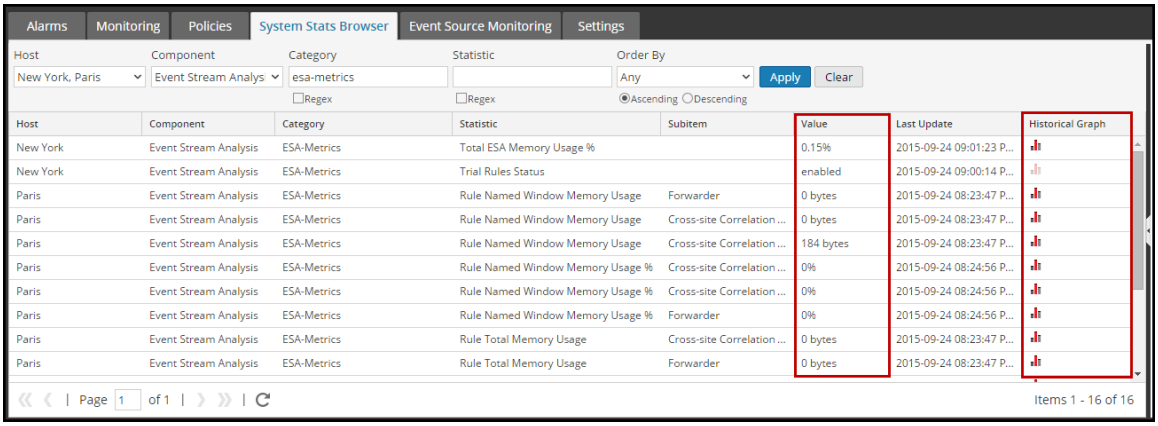
If your host is down, contact your NetWitness Suite Administrator to restart it. Otherwise, go to Step 2.

Step 2: View Detailed Statistics in Health & Wellness

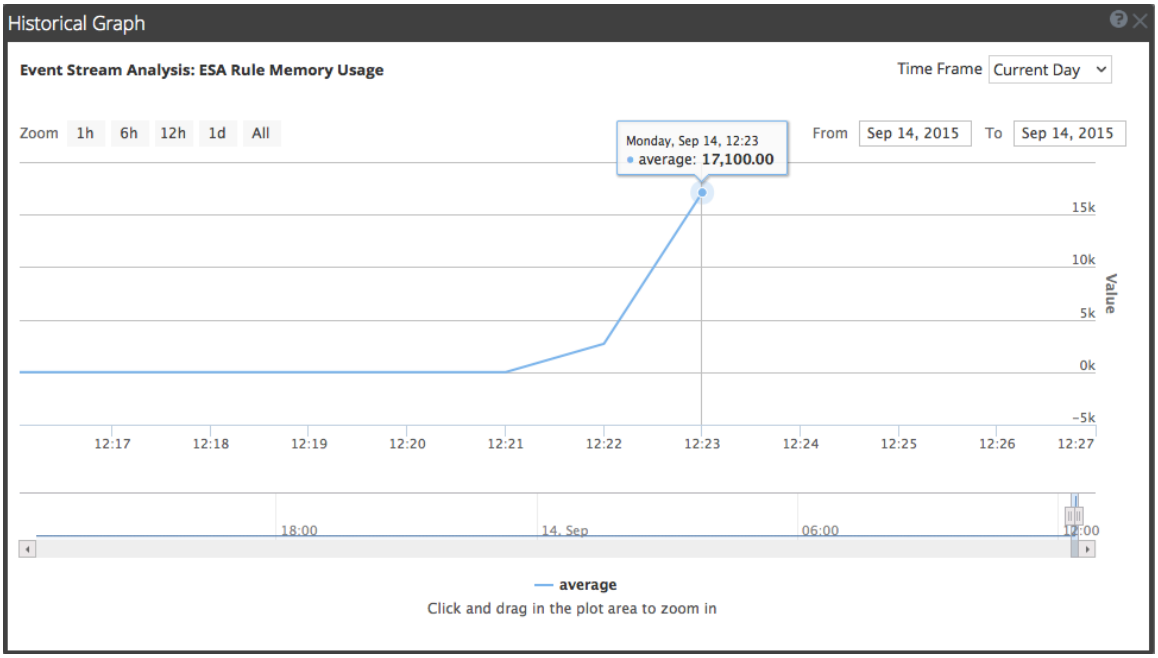
Once you are sure your ESA service is down, you can go to Health & Wellness to see where potential issues are occurring. The most common problem is that your ESA service is exceeding memory thresholds, which causes it to stop or fail.

1. Go to **ADMIN** > **Health & Wellness** > **Alarms** to see if the ESA triggered any alarms. Look for the following alarms:
   - ESA Overall Memory Utilization > 85%
   - ESA Overall Memory Utilization > 95%
   - ESA Service Stopped
2. Go to **ADMIN** > **Health & Wellness** > **System Stats Browser** to see the memory metrics for each rule's performance. To view the metrics, enter the following:

Host	Component	Category
<your host>	Event Stream Analysis	esa-metrics



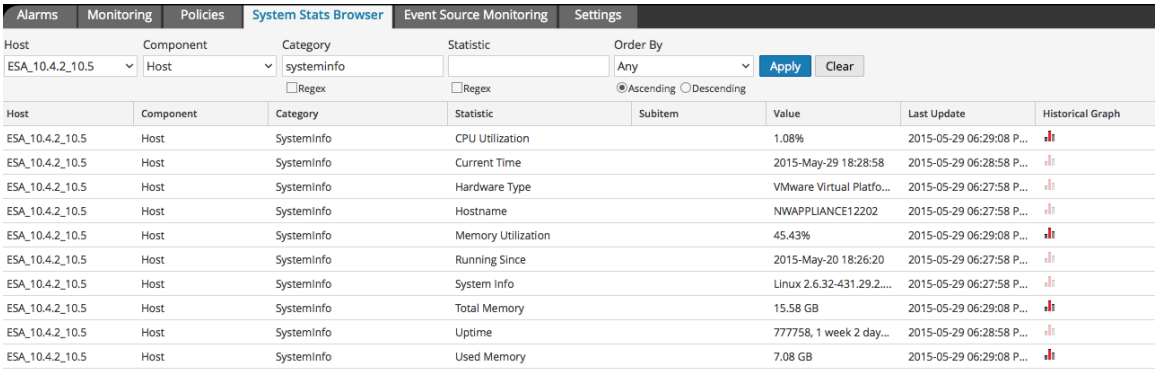
The memory for each rule is displayed in the **Value** column, and the value is displayed in bytes. You can view a historical view of memory usage in the **Historical Graph** column.



3. Go to **ADMIN** > **Health & Wellness** > **System Stats Browser** to see details of your ESA performance. Select your host, and use the following filters to view the following statistics:

Host	Component	Category	Statistic	Example
<your host>	Host	SystemInfo	CPU Utilization	1.08%
<your host>	Host	SystemInfo	Memory Utilization	45.43%
<your host>	Host	SystemInfo	Used Memory	7.08 GB
<your host>	Host	SystemInfo	Total Memory	15.58 GB
<your host>	Host	SystemInfo	Uptime	77758, 1 week, 2 day...
<your host>	Event Stream Analysis	ProcessInfo	Memory Utilization	7.07 GB

Host	Component	Category	Statistic	Example
<your host>	Event Stream Analysis	ProcessInfo	CPU Utilization	0.2%
<your host>	Event Stream Analysis	JVM.Memory	all	Committed Heap Memory Usage 8.0 GB
<your host>	Event Stream Analysis	ESA-Metrics	Total ESA Memory Usage %	4.64%



If you are having a problem with memory or CPU utilization, continue to step 3.

Step 3: Bring up your ESA Services

1. From **ADMIN** > **Services**, select the actions icon  for your ESA service and choose **start**.
2. Return to the ESA Service to troubleshoot which rules have created memory issues.

If your ESA service is stopping and restarting in a loop, you may need to call Customer Support to get the services to start.

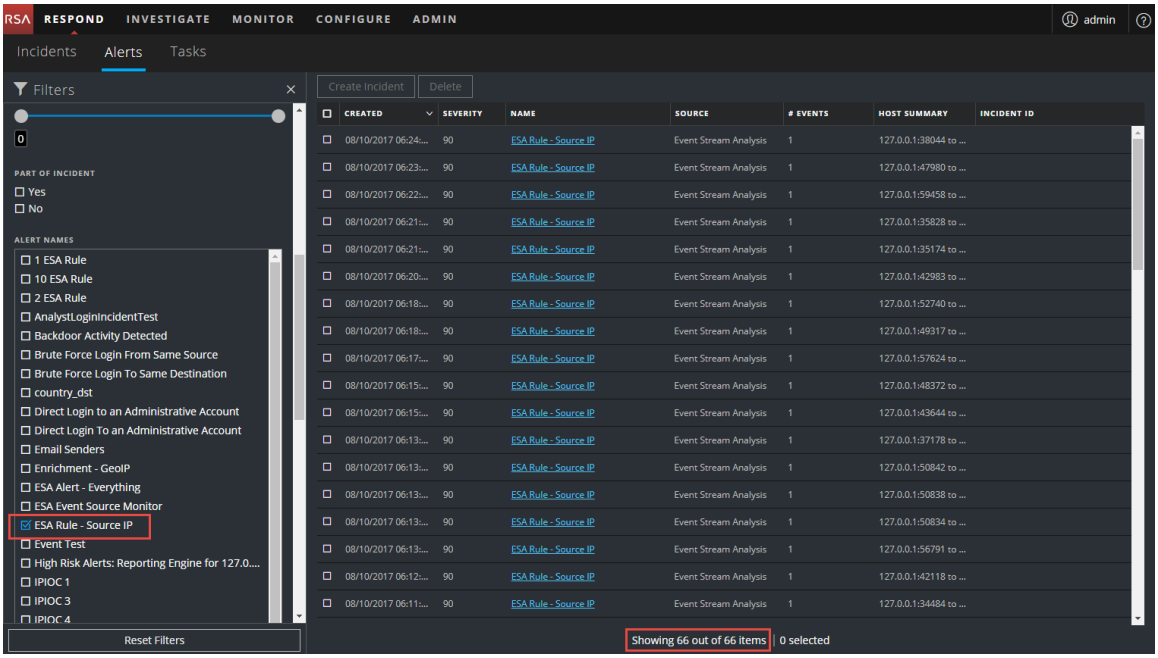
If you are able to start your ESA service without a shutdown, continue to step 4.

Step 4: Check the Alerts and Events Volume

Once you are able to restart your ESA service without an immediate shutdown, you can review the stats for your rules to see which rules are consuming too many resources. Sometimes, ESA services fail because a rule is generating too many alerts or a rule is matching too many events. Check for both of these issues if you have determined that memory usage is causing your ESA service to shut down.

View Alert Summaries

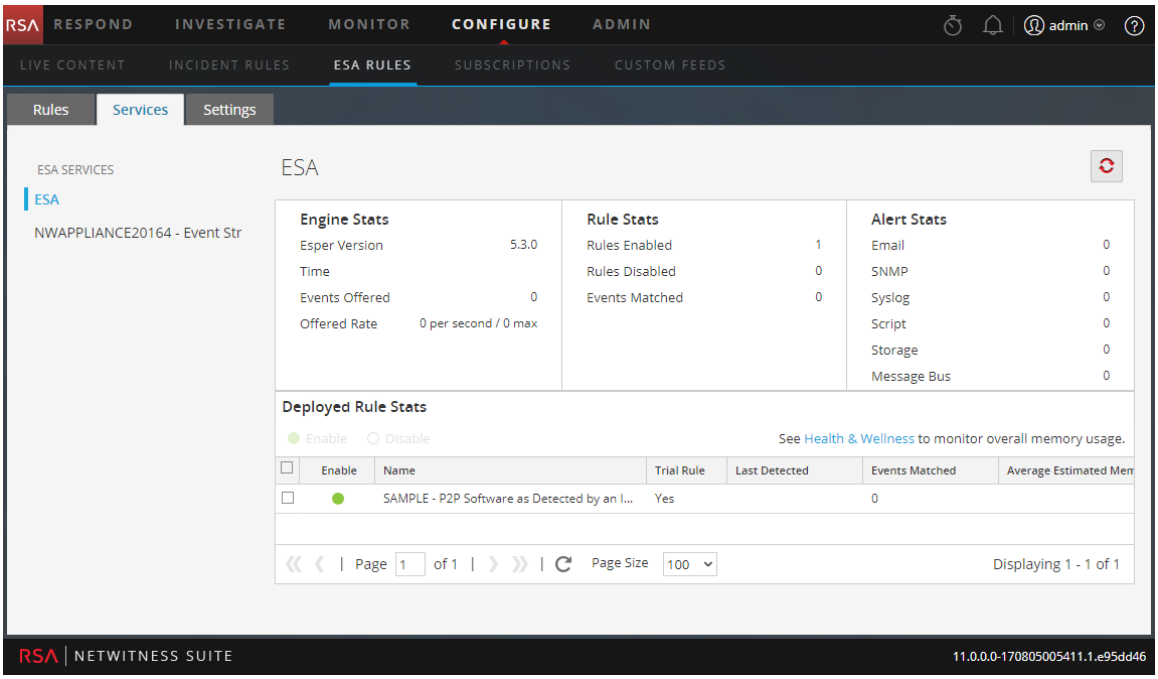
Rules that generate a high volume of alerts can overwhelm the system and cause it to fail or restart. To view the alert summaries, go to **RESPOND > Alerts**. In the **Filters** panel on the left, in the **ALERT NAMES** section, select the alert name for the rule. The number of alerts with that name appears at the bottom of the Alerts list results. If the number is significantly high for a particular rule, you need to disable the rule and rewrite it to be more efficient.



To clear your filter, click **Reset Filters**.

View Events Matched

Sometimes a rule matches too many events, which can use up excessive memory. This typically occurs if you create a large event window where a great number of events accumulates without triggering an alert. These are a problem because each event is stored in memory while the rule waits for the alert to trigger. To check for this issue, go to **CONFIGURE > ESA Rules > Services.** From there, you can see the number of events that were matched in the **Events Matched** column. If there was a high number of events matched for a given rule, you can investigate the rule further to see if you can make it more efficient.



Step 5: Disable and Repair the Rule that Caused Issues

Once you have determined the rules that need to be rewritten, disable them and rewrite rules so that they don't generate such a high volume of alerts or events. For pointers on how to write more efficient rules, see Best Practices.

Disable Rules

1. To disable rules, go to **CONFIGURE** > **ESA Rules** > **Services**, and select the rules you want to disable in the **Deployed Rules Stats** field.
2. Select **Disable** to disable the rules.

Edit Rules

1. To repair the rules, go to **CONFIGURE** > **ESA Rules** > **Rules** > **Rule Library**. Select the rule to edit, and click the actions icon  .
2. Select **Edit**.
3. Edit the rule to be more efficient. For instructions on creating rules, see Add Rules to the Rule Library
4. Once you are satisfied with your rule, you can save the rule as a trial rule to ensure that any memory issues do not affect ESA services performance. To do this, follow the steps listed in Work with Trial Rules.

Enable Rules

1. To enable rules, go to CONFIGURE > **ESA Rules** > **Services**, and select the rules you want to enable in the **Deployed Rules Stats** field.
2. Select **Enable** to enable the rules.

(Optional) Check the ESA Log Files for More Information

Once you verify that your services are down and some potential causes for the system going down, check to see if the service is stopping and restarting in a loop. To do this, go to the ESA logs. From the **ADMIN** > **Services** view, select your ESA service, and then select  > **View** > **Logs**.

If you cannot access the ESA logs from the NetWitness Suite interface, you can use SSH to get in the system and go to:`opt/rsa/esa/logs/esa.log`.

View Memory Metrics for Rules

This topic tells ESA rule writers how to view memory metrics for rules. You can see estimated memory usage for each rule running on a server, and you can use this information to modify your rule statements and conditions if they use too much memory.

Rules can sometimes consume more memory than you expect, causing your ESA to slow down or stop. To see approximately how much memory a rule is using, you can configure memory metrics. Memory metrics allow you to view an estimated memory usage for each rule in the Health & Wellness System Stats browser (so you will need permissions to access this module). You can use this information to modify your rules to be more efficient.

At a high level, you will need to complete the following steps to use the memory metrics to troubleshoot memory usage for rules:

1. Ensure that the memory metrics feature is enabled (via Explorer > CEP > Metrics > EnableStats). The Memory Metrics feature is enabled by default.
2. Ensure you have the correct permissions to view the Health & Wellness module. For information on roles and permissions, see Role Permissions.
3. View the memory statistics in Health & Wellness.
4. (Recommended) Configure Health & Wellness ESA policies to send an email if memory thresholds are exceeded. See "Manage Policies" in the *System Maintenance Guide* for instructions on sending email notifications.
5. Use the memory metrics data to modify rules to be more efficient, if necessary.

Prerequisites

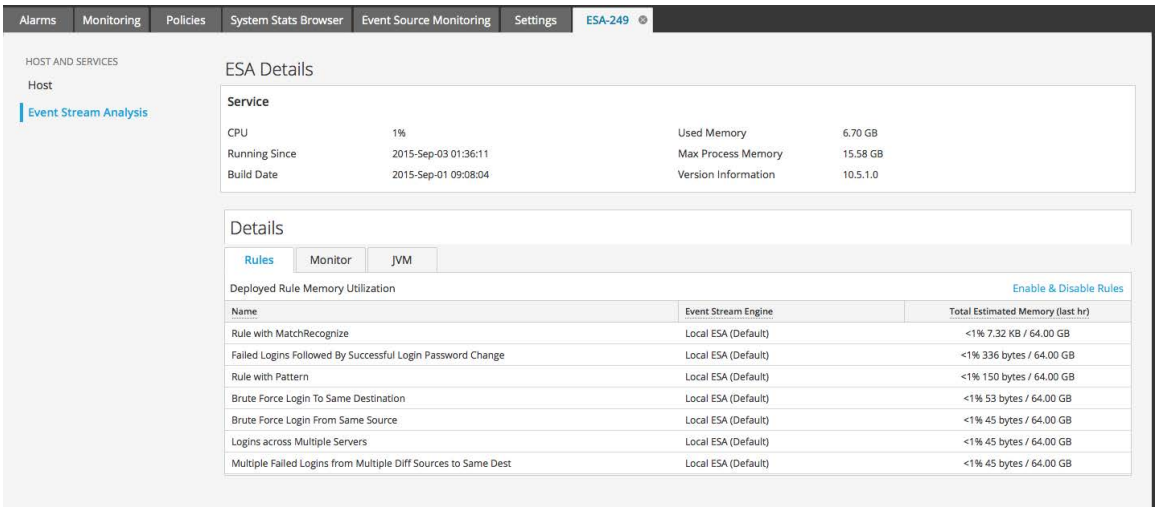
The following are requirements for using memory metrics:

- Memory Metrics feature is enabled (via **Explorer** > **CEP** > **Metrics** > **EnableStats**).
- The user must have the appropriate permissions to view Health & Wellness statistics.
- (Recommended) Configure the ESA Health & Wellness policy to send an email when memory thresholds are exceeded.

Procedures

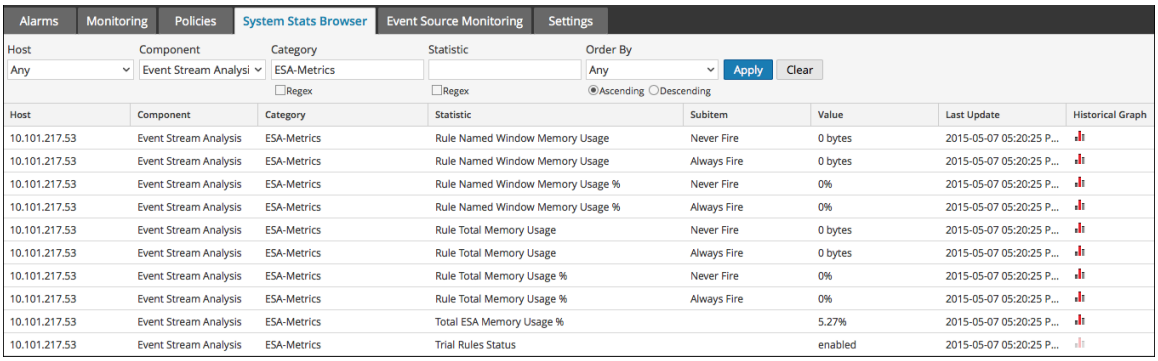
View Memory Metrics in the Health & Wellness System Monitoring Module

1. Go to **ADMIN** > **Health & Wellness** > **Monitoring**
2. View the details for your ESA service.
3. Click the **Rules** tab.
4. You can view the average memory usage for each rule for the previous hour.



View Memory Metrics in the Health & Wellness System Stats Browser

1. Go to **ADMIN** > **Health & Wellness** > **System Stats Browser**.
2. For component, select **Event Stream Analysis**. For category, enter **ESA-Metrics**.



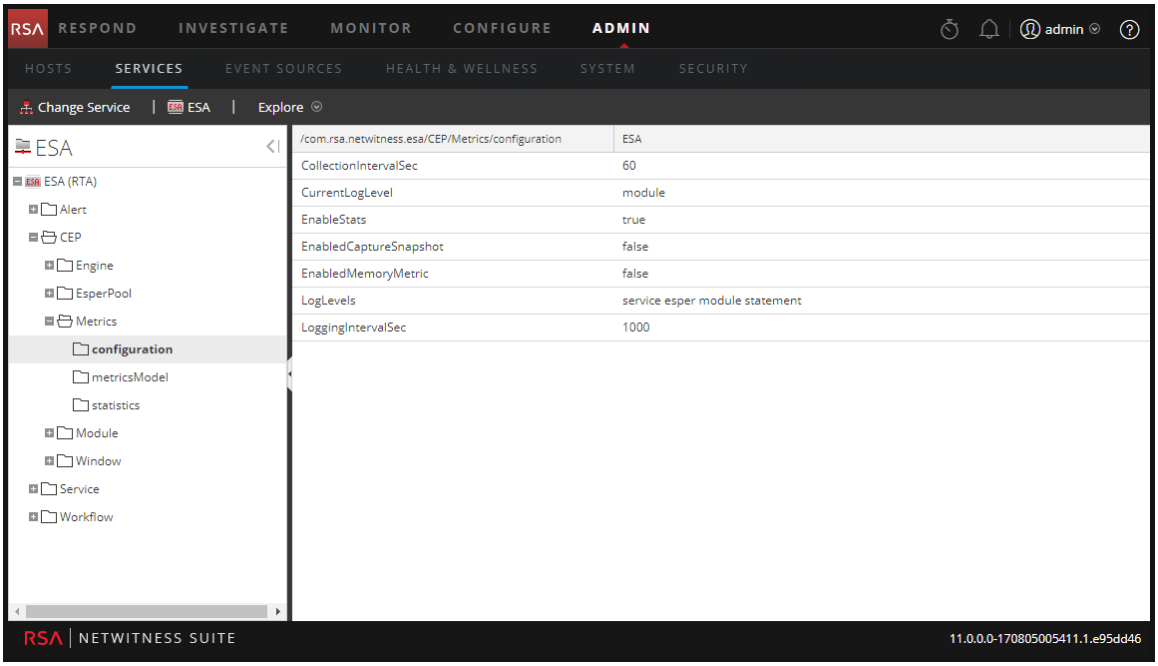
The name of the rule is displayed in the **Subitem** field, and the memory usage is displayed in the **Value** column.

3. To view the historical memory usage for the rule, click on the **Historical Graph** icon.

Note: The **Last Update** field reflects when Health & Wellness polls ESA. However, the Memory Metrics is not synchronized with the Health & Wellness polling. For example, if the memory threshold is exceeded on 10/10/15 at 12 p.m., but Health & Wellness polls at 10/10/15 at 12:10 p.m., the **Last Update** field will display a timestamp of 10/10/15 12:10 p.m.

Enable or Disable the Memory Metrics Feature

1. Go to **ADMIN** > **Services**and select your ESA.
2. Once you've selected your ESA, click **Actions** > **View**> **Explore**, and navigate to **CEP** > **Metrics** > **Configuration** as shown below.



3. Change the field EnabledStats to **true** or **false** depending on whether you want to enable or disable the memory metrics feature.

How ESA Generates Alerts

This topic provides a brief description of how an Event Stream Analysis (ESA) service runs rules to generate alerts. The Event Stream Analysis (ESA) service runs rules that specify criteria for problem behavior or threatening events in your network. When ESA detects a threat that matches rule criteria, it generates an alert.

To generate alerts, ESA performs the following functions:

1. Gathers data
2. Runs ESA rules against the data
3. Captures events that meet rule criteria
4. Generates alerts for those captured events

You can use the Alerts module to gain visibility into your network and to detect problems in it.

Sensitive Data

This topic explains how ESA treats sensitive data, such as usernames or IP address, that it receives from Core services. The Data Privacy Officer (DPO) role can identify meta keys that contain sensitive data and should display obfuscated data. ESA will not display or store sensitive meta. Consequently, ESA will not pass sensitive data to NetWitness Respond.

Optionally, ESA can add an obfuscated version of the sensitive data to an event. For example, the DPO identifies user_dst as sensitive. ESA can add an obfuscated version, such as user_dst_hash, to an event. The obfuscated meta is not sensitive, so ESA will display and store it the same way as any other non-sensitive meta.

For more information on the strategy and benefits of obfuscating data, see the *Data Privacy Management Guide*.

This topic explains the following:

- How ESA treats sensitive data it receives from Core services
- How to prevent sensitive data leaks in an Advanced EPL rule

How ESA Treats Sensitive Data from Core Services

When ESA receives sensitive data from Core services, ESA passes on only the obfuscated version of the data. ESA does not store or show sensitive data.

The following features are impacted:

- Outputs – ESA does not forward sensitive data to outputs, which include alerts, notifications and MongoDB storage.
- Advanced EPL rules – If an EPL statement creates an alias for a sensitive meta key, sensitive data will leak. This topic illustrates how this happens so you can avoid it.
- Enrichments – If a sensitive meta key is used in the join condition, sensitive data will leak. This topic illustrates how this happens so you can avoid it.

Advanced EPL Rule

If an EPL query statement renames a sensitive meta key, the data will not be protected.

ESA identifies a sensitive meta key by the name:

ip_src is the sensitive meta key.
ip_src_hash is the non-sensitive, obfuscated version.

To support data privacy, the sensitive meta key must not be renamed in an EPL query. If a sensitive meta key is renamed, the data will no longer be protected.

For example, in a rule such as select ip_src as ip_alias..., ip_alias contains the sensitive data but it is not protected because ESA only knows about ip_src, not ip_alias. In this case, IP addresses would not be obfuscated. Real values would be displayed.

Enrichment Source

When a sensitive meta key is used in a join condition, sensitive data can be displayed.

The enrichment database, which is the other part of the join condition, has one column that matches the sensitive meta key. This cross reference is to actual values not obscured values. Consequently, actual values are displayed.

In the following example, both parts of the join condition are highlighted.

- ip_src contains sensitive data.
- ipv4 will be added to the alert and exposed as non-sensitive data

Because the ipv4 value is the same as the ip_src value, ipv4 contains and displays sensitive data.

ESA Rule Types

This topic describes each type of ESA rule, when to use them and the permissions each role has with them. The following table lists each type, describes it, and explains when to use it.

Rule Type	Description	When to Use
Rule Builder	In the rule builder, you define rule criteria in an easy-to-use interface.	Use the rule builder to create your first rules. You choose many of the rule conditions from lists.
Advanced EPL	With the Event Processing Language (EPL), you define rule criteria by writing a query.	Use advanced EPL rules to define rule criteria in the EPL syntax.
RSA Live ESA	RSA Live has a catalog of ESA rules that you can download and modify to run in your network.	Download RSA Live ESA rules to leverage rules that are already built. Modify the configurable parameters to customize to meet your requirements.

Starter Pack Rules

A few sample Rule Builder rules come with NetWitness Suite and appear in the Rule Library. Use starter pack rules to get comfortable working with rules before creating your own. You can safely edit and deploy these sample rules.

Trial Rules Mode

For any type of rule, you can select the Trial Rule setting as an additional safeguard. Trial rules get disabled if they exceed a memory threshold the administrator sets. Run a rule in trial mode to monitor memory usage and to disable the rule automatically if it uses more memory than the threshold allows.

Role Permissions

This topic lists all ESA permissions and shows which permissions are assigned to each pre-configured NetWitness Suite role. User access is restricted based on roles and permissions assigned to roles.

- Administrators
- Operators
- Analyst
- Security Operations Center (SOC) Managers
- Malware Analysts (MA)
- Data Privacy Officer

There are four permissions for ESA:

1. Access Alerting Module – Is required for any permission
2. View Rules – Allows view-only permission for rules in the Rule Library
3. View Alerts – Allows view-only permission for alerts ESA generates
4. Manage Rules – Allows you to view, create, edit and delete rules

The following table lists permissions for ESA and the roles to which they are assigned. Use this table to see how each role can work with rules and alerts.

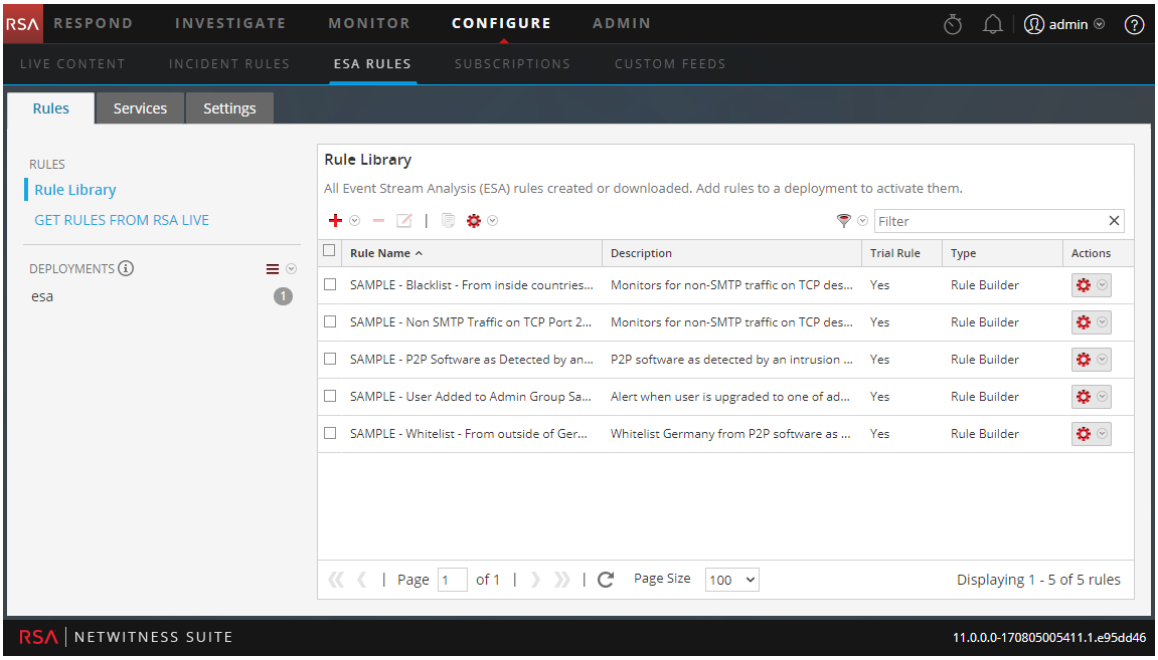
Permission	Administrators	Operators	Analysts	SOC Mgrs	MAs	DPOs
Access Alerting Module	Yes	Yes	Yes	Yes		Yes
View Rules	Yes	Yes		Yes		Yes
View Alerts	Yes		Yes	Yes		Yes
Manage Rules	Yes	Yes		Yes		Yes

For more information on roles and permissions, see the *System Security and User Management Guide*.

Practice with Starter Pack Rules

NetWitness Suite comes with starter pack rules so analysts can become familiar with how rules look before you create your own rules. Use the starter pack rules to become familiar with the Rule Builder and to practice editing and deploying a rule.

Starter pack rules are installed in the Rule Library, which will contain every rule you download or create. The following figure shows sample rules in the Rule Library.



These are the available starter pack rules:

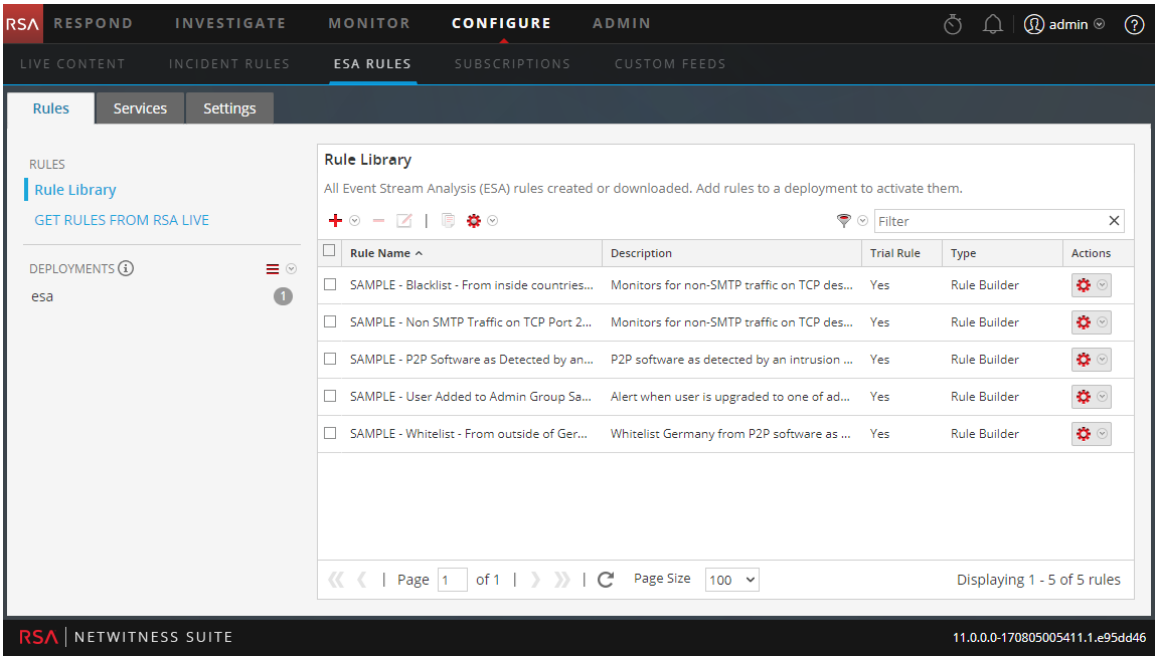
- SAMPLE: P2P Software as Detected by an Intrusion Detection Device
- SAMPLE: Non SMTP Traffic on TCP Port 25 Containing Executable
- SAMPLE: Whitelist - From outside of Germany, P2P Software as Detected by an Intrusion Detection Device.
- SAMPLE: Blacklist - From inside countries that are not the US, Non-SMTP Traffic on TCP Port 25 Containing Executable
- SAMPLE: User Added to Admin Group Same User su Sudo

Each name begins with SAMPLE to distinguish the rules that are installed with NetWitness Suite from the rules you download and create.

Rule Library

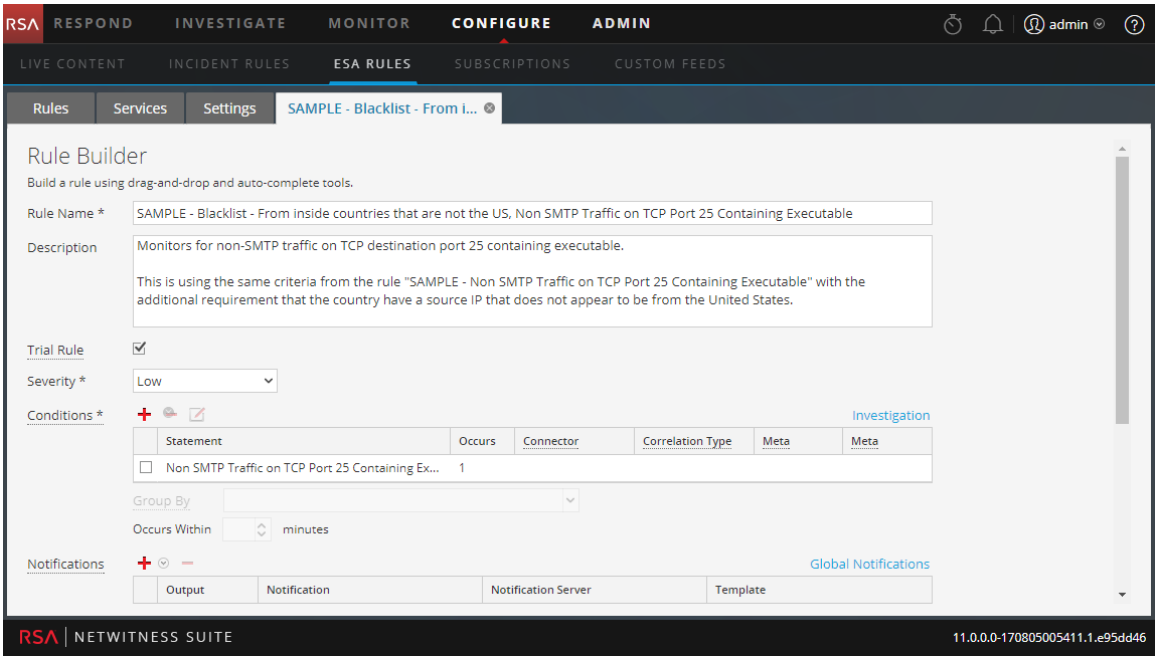
The Rule Library shows the following information for a rule:

- **Name** summarizes the data or events the rule collects.
- **Description** explains the rule in more detail, although only the beginning shows in the Rule Library.
- **Trial Rule** indicates if trial mode is enabled or disabled for the rule.
- **Type** shows the origin of the rule, built in Rule Builder or Advanced EPL, or downloaded from RSA Live.



Procedure

1. Go to **CONFIGURE > ESA Rules**.
   The ESA Rules view is displayed with the Rules tab open.
2. In the **Rule Library**, select a sample rule and click , or double-click a rule.
   The rule is opened in Rule Builder.



3. To practice with a starter pack rule, refer to the following topics for detailed descriptions and procedures:
   - To familiarize yourself with the Rule Builder user interface, see Rule Builder Tab for a description of each field.
   - To learn how to edit a rule, see Add a Rule Builder Rule for a step-by-step procedure.
   - To deploy a starter pack rule, see Deploy Rules to Run on ESA to learn how to associate the rule with an ESA service.

After you practice with starter pack rules, you will be able to download, create, and deploy your own rules.

Work with Trial Rules

When rules use too much memory, your ESA service can become slow or unresponsive. To ensure rules do not use excessive memory, you can enable trial rules for any type of rule. By default, new rules you create and RSA Live rules you import are configured to be trial rules. RSA recommends you disable the trial rule setting only after testing the new rule in your environment during normal and peak network traffic. When you create a trial rule, you set a global threshold of the percentage of memory that rules may use. If that configured memory threshold is exceeded, all trial rules are disabled.

The NetWitness Suite Event Stream Analysis (ESA) service is capable of processing large volumes of disparate event data from Concentrators. However, when working with Event Stream Analysis, it is possible to create rules that use excessive memory. This can slow your ESA service or even cause it to shut down unexpectedly. To ensure that this doesn't happen, you can configure your rule as a trial rule. When you configure a trial rule, you also set global threshold of the percentage of memory that rules may use. If that configured memory threshold is exceeded, all trial rules are disabled automatically.

For suggestions on creating more efficient rules, see "Best Practices for Writing Rules" in Best Practices

By default, new rules and RSA Live rules are configured as trial rules. As a best practice, when you edit an existing rule, select the Trial Rule option, which allows you to:

- Deploy the rule with an added safeguard.
- Optionally, view a snapshot of memory utilization to understand if the rule creates memory issues.
- Know if you must modify the rule criteria to improve performance.

Note: Run a rule as a trial rule long enough to assess the performance during normal and peak network traffic.

Deploy Rules as Trial Rules

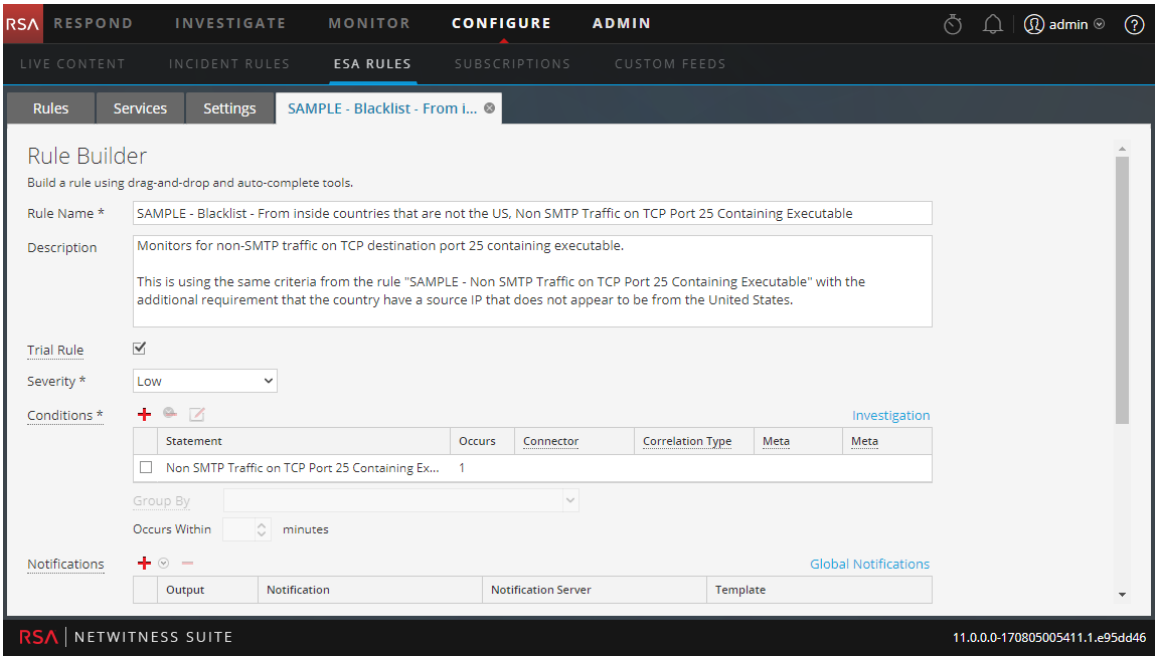
This topic explains to administrators how to enable trial rules when creating new rules or editing rules. Trial rules are automatically disabled if a specified total JVM memory utilization threshold is exceeded.

Procedure

To deploy rules as trial rules:

1. Go to **CONFIGURE** > **ESA Rules**.
   The Configure ESA Rules view is displayed with the Rules tab open.

2. From the Rule Library, choose to add or edit a rule. The rule builder is displayed in a new tab.



3. To make a new or existing rule a trial rule, select the **Trial Rule** checkbox.
4. Add the rule conditions or modify the rule as needed. For instructions on editing rules, see Add Rules to the Rule Library.
5. Click **Save**.
6. Ensure that trial rules are enabled for your ESA and that you are satisfied with the thresholds configured for trial rules.
   The memory threshold is set in the configuration file. To configure it, see "Change Memory Threshold for Trial Rules" in the *ESA Configuration Guide*.
   The threshold is configured per ESA and is a percentage of Java Virtual Memory.
   The configuration parameter, MemoryThresholdforTrialRules default is 85.
7. Optionally, you can set up the policies in Health and Wellness to send you an email notification if the total JVM memory utilization threshold is exceeded.

The next time you deploy the rule, it runs in trial rule mode.

Note: If a trial rule is disabled, you will need to go to the **CONFIGURE** > **ESA Rules** > **Services** tab to re-enable the trial rules. For more instructions on re-enabling trial rules on a service, see View ESA Stats and Alerts.

View Memory Metrics for Rules Using Trial Mode

This topic tells ESA rule writers how to view memory metrics when the memory threshold configured for trial rules is exceeded. If the memory threshold is exceeded, you can configure a snapshot to be taken of the memory usage for ESA rules at the time that trial rules are disabled, allowing you to investigate memory usage and edit the rules to be more efficient.

When you configure trial rules and enable the Memory Snapshot feature, if the memory threshold is exceeded, all trial rules are disabled and a snapshot of the memory usage for all ESA rules is taken at the time of disablement. This allows you to see how much memory was used so that you can modify your ESA rules to be more efficient. The memory snapshot can be viewed in the Health & Wellness System Stats browser, so you will need permissions to access this module. Once you view the details in the System Stats browser, you can modify the trial rule syntax and re-enable the trial rules.

At a high level, you will need to complete the following steps to use the Memory Snapshot to troubleshoot memory usage for rules:

1. Enable trial rules for any new rules you deploy. See Deploy Rules as Trial Rules.
2. Ensure that you have configured Health & Wellness ESA policies to send an email if memory thresholds are exceeded.
3. Ensure you have the correct permissions to view the Health & Wellness module. For information on roles and permissions, see Role Permissions.
4. Ensure that the Memory Snapshot feature is enabled (via the EnabledCaptureSnapshot parameter via NetWitness Suite Explorer). The Memory Snapshot feature is disabled by default. See "Enabling and Disabling the Memory Snapshot Feature" below. RSA recommends you disable the feature once you have completed testing new rules.
5. View the memory threshold statistics in Health & Wellness if the memory threshold is triggered for trial rules.
6. Modify the rule or rules that triggered the alarm. For best practices for rule writing, see Best Practices.
7. Re-enable the trial rules that were disabled when the memory threshold was triggered. For instructions on re-enabling trial rules on a service, see View ESA Stats and Alerts.
8. Continue to test the trial rules.

Note: Like any Debug tool, there can be exceptional overhead associated with using the Memory Snapshot feature. When actively taking a snapshot, the Memory Snapshot feature can add delays to your ESA services. The ESA service stops generating alerts while taking a snapshot. RSA recommends you disable the feature once you have completed testing new rules. If you disable the Memory Snapshot feature, trial rules will still be disabled when memory usage exceeds configured thresholds, but the memory snapshot will not be taken, and the statistics will not appear in the Health & Wellness System Stats browser.

Prerequisites

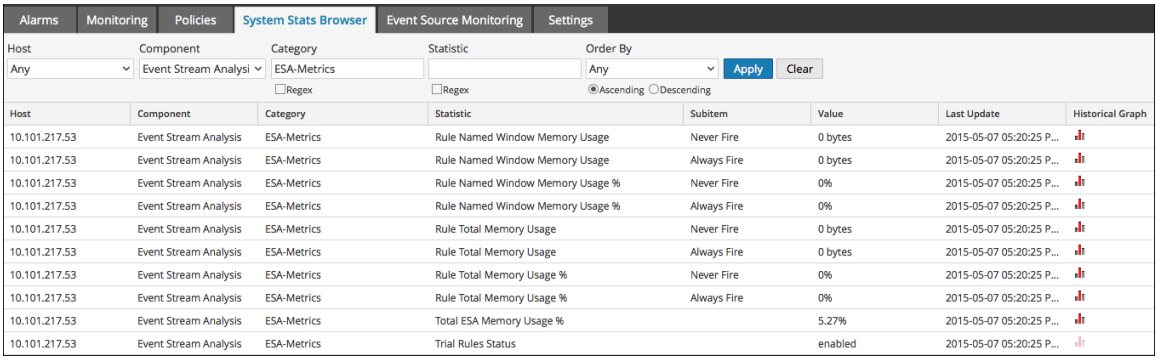
These are the requirements for viewing memory metrics:

- One or more ESA rules must be configured as a trial rule.
- Memory Snapshot must be enabled (via the EnabledCaptureSnapshot parameter via NetWitness Suite Explorer).
- The user must have the appropriate permissions to view Health & Wellness statistics.
- The user must have configured the ESA Health & Wellness policy to send an email when memory thresholds are exceeded.

Procedures

View Memory Metrics

1. Go to **ADMIN** > **Health & Wellness** > **System Stats Browser**.
2. For component, select **Event Stream Analysis**. For category, enter **ESA-Metrics**.

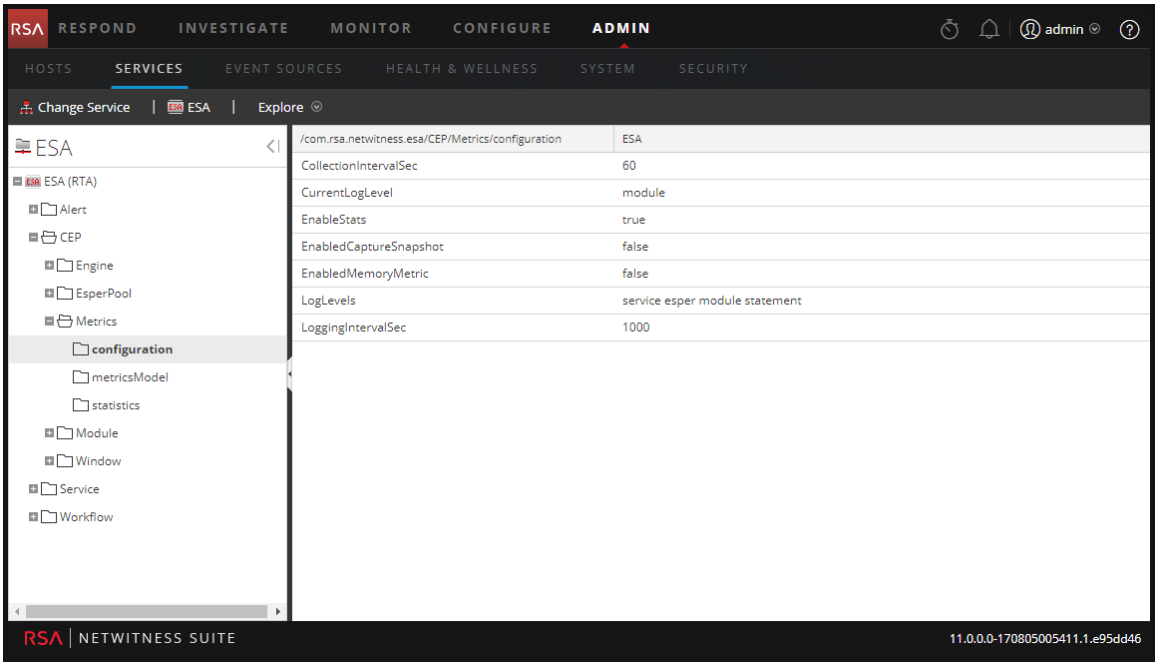


The name of the rule is displayed in the **Subitem** field, and the memory usage is displayed in the **Value** column.

Note: The **Last Update** field reflects when Health & Wellness polls ESA. However, the Memory Snapshot only occurs when memory thresholds are exceeded, so this field does not reflect when the snapshot was taken or updated. The snapshot remains static until the memory threshold is exceeded again. For example, if the memory threshold is exceeded on 10/10/15 at 12 p.m., but Health & Wellness polls at 10/10/15 at 3 p.m., the **Last Update** field will display a date of 10/10/15 3 p.m.

Enable or Disable the Memory Snapshot Feature

1. Go to **ADMIN** > **Services**and select your ESA.
2. Select  > **View** > **Explore**, and navigate to CEP > Metrics > Configuration as shown below.



3. Change the field EnabledCaptureSnapshot to **true** or **false** depending on whether you want to enable or disable the Memory Snapshot feature.

Add Rules to the Rule Library

This topic explains how to add each type of rule to the rule library. You must add a rule to the Rule Library before you can deploy it. Permission to manage rules is required for all tasks in this section. To add rules, you can download them from ESA Live, create a rule via the Rule Builder, or write advanced EPL rules.

For more details on each of these procedures, see:

- Download Configurable RSA Live ESA Rules
- Add a Rule Builder Rule
- Add an Advanced EPL Rule

In addition to deploying a rule, you can edit, duplicate, import, export, and remove a rule in the Rule Library. For details on these procedures, see Working with RulesWorking with Rules

Download Configurable RSA Live ESA Rules

This topic explains how to download configurable rules from the NetWitness Suite Live Content Management System so you can customize them to meet your needs.

RSA Live contains a catalog of rules. Each rule has configurable parameters so you can customize the rule for your environment. If RSA Live has a rule to detect events that you want to detect in your network, download the rule to save time. You can edit the configurable parameters and save the rule in your Rule Library.

This is a sample of how each RSA Live ESA rule is described on RSA Live:

Rule Name	Description
Logins across Multiple Servers	Detects logins from the same user across 3 or more separate servers within 5 minutes. The time window and number of unique destinations are configurable.

As the name shows, the rule looks for logins across multiple servers. The description explains the rule criteria in more detail and specifies which parameters you modify.

Note: When a rule description includes a configurable parameter, the default setting for the parameter is used. In the sample rule, the description states 5 minutes. However, the time window is configurable so 5 is the default number of minutes.

Prerequisites

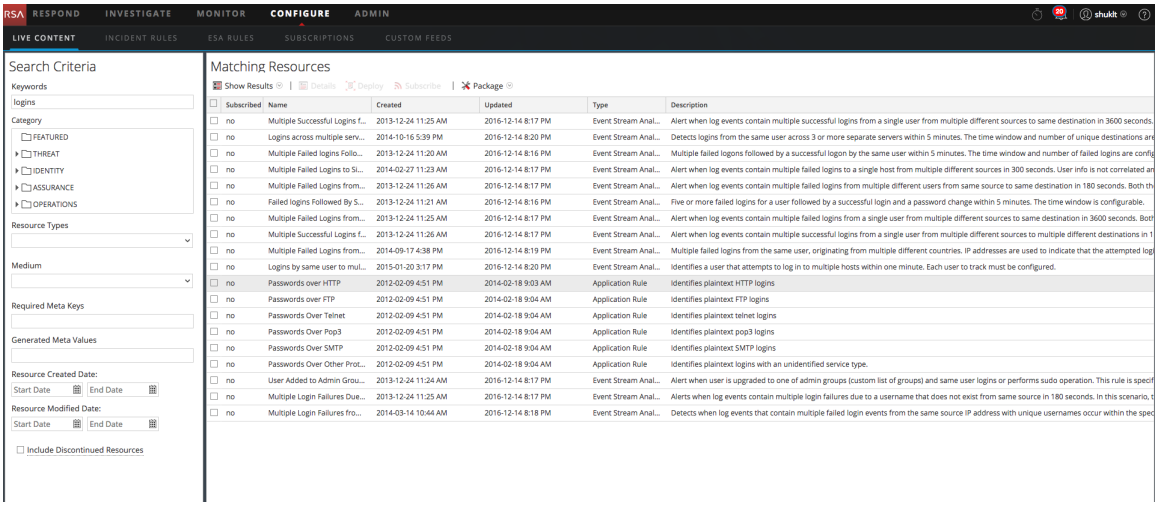
These are the prerequisites for downloading configurable RSA Live ESA rules;

- Have permission to manage rules
- Create a Live Account. See the *Live Services Management Guide* for details.
- Set up Live on NetWitness Suite. See the *Live Services Management Guide*for details.

Procedure

To download configurable RSA Live ESA rules:

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab is displayed.
2. In the options panel, click **Get Rules from RSA Live**.
   The Live Content Search view is displayed.



3. In **Search Criteria**, for **Resource Type** select **RSA Event Stream Analysis Rule**.
4. Specify any of the following criteria to find a rule to configure for your environment.
   For a detailed description of the search criteria, see "The Live Search View" in the *Live Services Management Guide*.
   a. Keywords
   b. Tags
   c. Required Meta Keys
   d. Generated Meta Values

e. Resource Created Date

f. Resource Modified Date

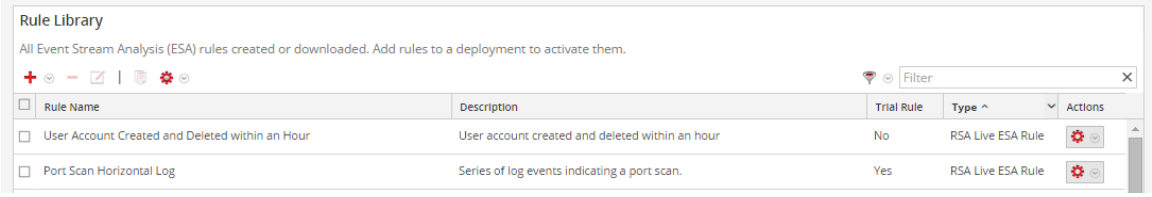
5. Click **Search**. Rules that match the search criteria are displayed in Matching Resources.
6. Select each rule to download and click **Deploy**.
   The Deployment Wizard is displayed
7. Follow the steps in the wizard. If you need more information, see "Deploy Resources in Live" in the *Live Services Management Guide*.

When you finish the steps in the wizard, the selected rules are displayed in the Rule Library.

Customize an RSA Live ESA Rule

This topic explains how to configure parameters in an RSA Live ESA rule. When you download an RSA Live ESA rule, the rule appears in the Rule Library which includes the following columns:

- Name
- Description
- Trial Rule
- Type



The type is RSA Live ESA Rule.

Prerequisites

- Administrator, Operator, SOC Manager, or DPO role permissions are required.
- Rules must be downloaded to the Rule Library.

Procedure

To customize an RSA Live ESA rule:

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
2. In the **Rule Library**, select an RSA Live ESA Rule and click .
   The RSA Live ESA Rule tab is displayed.

3. (Optional) Change the following fields:
   - Rule Name
   - Description
   - Trial Rule (Enabled by default. RSA recommends you run a rule as a trial rule long enough to assess the performance during normal and peak network traffic.)
   - Severity
4. To configure the rule for your environment, in the **Parameters** section replace the default in the **Value** Column.

5. Click **Save**

Add a Rule Builder Rule

This topic introduces a set of end-to-end procedures for adding a Rule Builder type rule.

Each ESA rule is designed to detect something in your network and to generate an alert for it:

- User activity that is not allowed, such as attempting to download software that is not sanctioned
- Suspicious behavior, such as mass audit clearing
- Known malicious threats, such as worm propagation or a password-cracking tool

There are two methods to design a rule in ESA:

- Rule Builder is an easy-to-use interface. You provide a meta key and value, then select choices from lists to complete the criteria.
- Advanced EPL allows you to write queries in the Event Processing Language. You must know EPL syntax.

If you know EPL, you can use either method. If you do not know EPL, you must use Rule Builder. These topics explain the Rule Builder.

Step 1. Name and Describe the Rule

This topic provides instructions to identify a rule, indicate if it is a trial rule and assign a severity level. When you add a new rule, the first information to provide is a unique name and description of what the rule detects. After you save the rule, this information is displayed in the Rule Library.

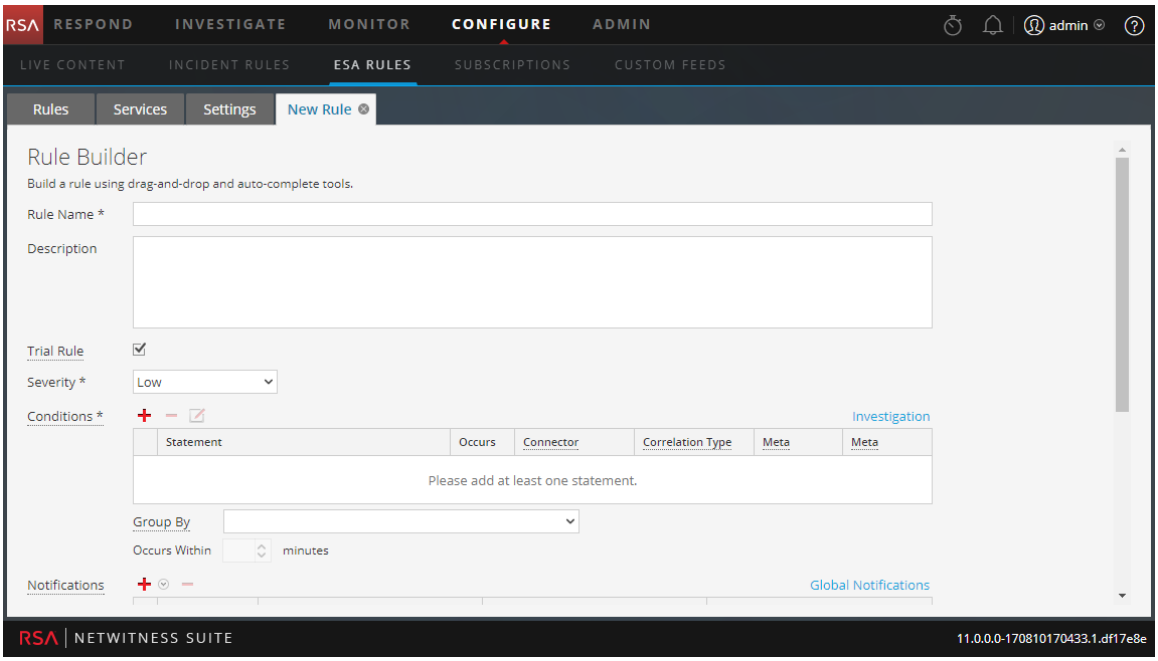
Prerequisites

You must have permission to manage rules. See Role Permissions.

Procedure

To name and describe a rule:

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
2. In the **Rule Library**, select + > **Rule Builder**.
   The New Rule tab is displayed.



3. Type a unique, descriptive name in the **Rule Name** field.
   This name will appear in the Rule Library so be specific enough to distinguish the rule from others.
4. In the **Description** field, explain which events the rule detects.
   The beginning of this description will appear in the Rule Library

5. By default, new rules are configured as a Trial Rule. A trial rule automatically disables the rule if all trial rules collectively exceed the memory threshold. If you are editing an existing rule, you can select **Trial Rule** to safely test the rule edits.
   Use trial rule mode as a safeguard to see if a rule runs efficiently and to prevent downtime caused by running out of memory. For more information, see Work with Trial Rules.
6. For **Severity**, classify the rule as Low, Medium, High or Critical.

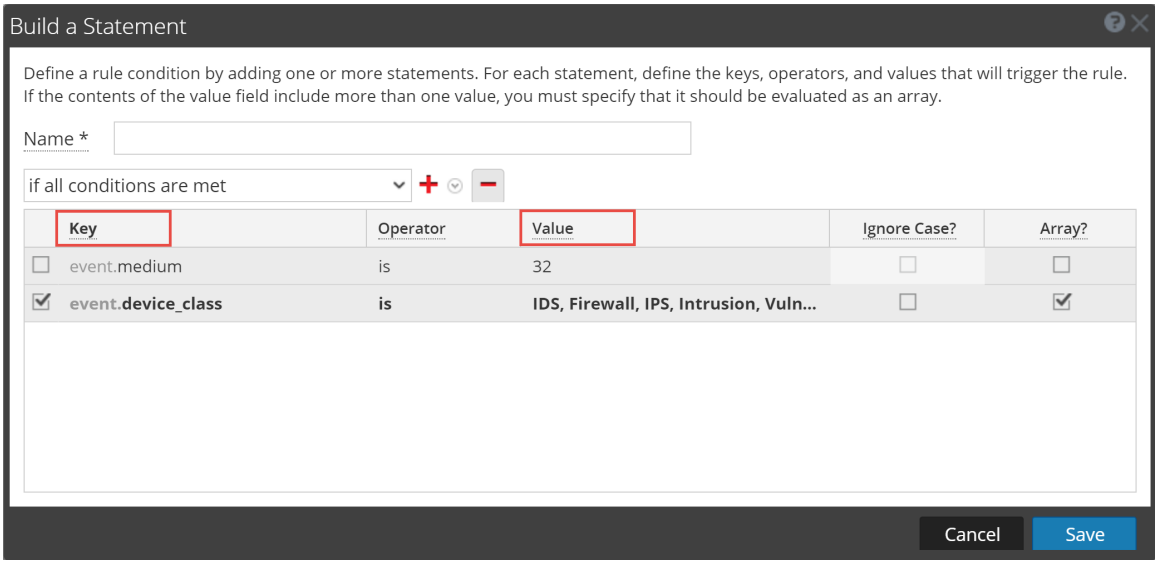
Step 2. Build a Rule Statement

This topic provides instructions to define rule criteria in Rule Builder by adding statements. A statement is a logical grouping of rule criteria in the Rule Builder. You add statements to define what a rule detects.

Example

The following graphic shows an example of a Rule Builder statement.

Every statement contains a key and value. Then, you build logic around the pair by selecting an option in each other field.



Prerequisites

To build a rule statement, you must know the meta key and the meta value.
For a complete list of meta keys, go to **CONFIGURE > ESA Rules > Settings > Meta Key References**.

Procedure

To build a rule statement:

1. Go to **CONFIGURE > ESA Rules**.

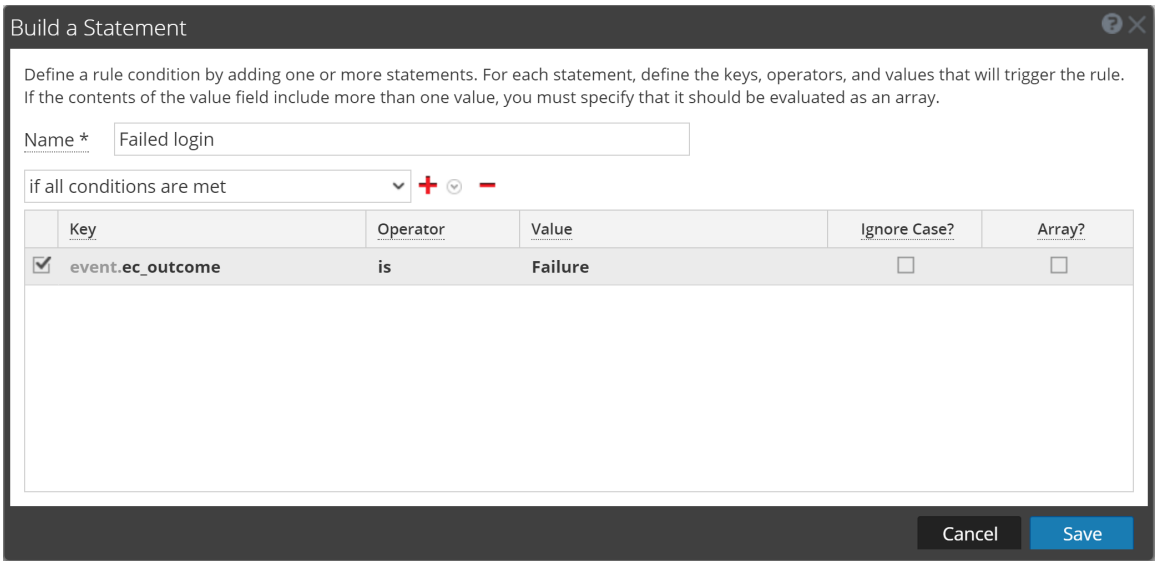
   The Rules tab is displayed by default.

2. In the **Rule Library**, click + > **Rule Builder** or edit an existing Rule Builder rule.

   The Rule Builder view is displayed.

3. In the **Conditions** section, click + .

   The Build Statement dialog is displayed.



4. **Name** the statement. Be clear and specific. The statement name will appear in the Rule Builder.
5. From the drop-down list, select which circumstances the rule requires:
   - if **all conditions** are met
   - if **one of these conditions** are met
6. Specify the criteria for the statement:
   a. For **Key,** type the name of the **Meta Key**.
   b. For **Operator** specify the relationship between the meta key and the value you will provide for it.
      The choices are: is, is not, is not null, is greater than (>), is greater than or equal to (>=), is less than (<), is less than or equal to (<=), contains, not contains, begins with, ends with
   c. Type the **Value** for the meta key.
      Do not add quotes around a value. Separate multiple values with a comma.

d. The **Ignore Case?** field is designed for use with string and string array values. By choosing the **Ignore Case** field, the query will treat all string text as a lowercase value. This ensures that a rule that searches for the user named Johnson would trigger if the event contains "johnson," "JOHNSON," or "JoHnSoN."

e. The **Array?** field indicates if the contents of the Value field represent one or more than one value.

   Select the Array checkbox if you entered multiple, comma-separated values in the **Value** field. For example, "ec_activity is Logon, Logoff" requires you to select the Array checkbox.

7. To use another meta key in the statement, click +, select **Add Meta Condition** and repeat step 6.
8. To add a whitelist, click + and select **Add Whitelist Condition**.
9. To add a blacklist, click + and select **Add a Blacklist Condition**.
10. To save the statement, click **Save**.

To Add a Whitelist

You use a whitelist to ensure that specified events are excluded from triggering the rule. Whitelists can be based on geographic location or by customer-defined enrichment CSV sources. For example, if you want to create a rule that only triggers for IP addresses outside of the US, you can create a whitelist of US IP addresses.

1. After you add a meta condition, click + and select **Add Whitelist Condition**.
2. In the **EnterWhitelist Name** field, select an enrichment source. Any enrichment source loaded from a CSV or a named window in Esper can be used as source for a whitelist.
3. If you used a GeoIP source for the whitelist, ipv4 is automatically entered for the subcondition. Enter the meta value for the corresponding value field. For example, enter *ipv4 is ip_src* to ensure the GeoIP records are selected based upon the ip_src being found in the GeoIP lookup database. In addition, if you used a GeoIP source for the whitelist, you might want to add a subcondition to specify the geographic region to exclude from the rule results. For example, to specify that the country code must be USA, enter "*CountryCode is US*".

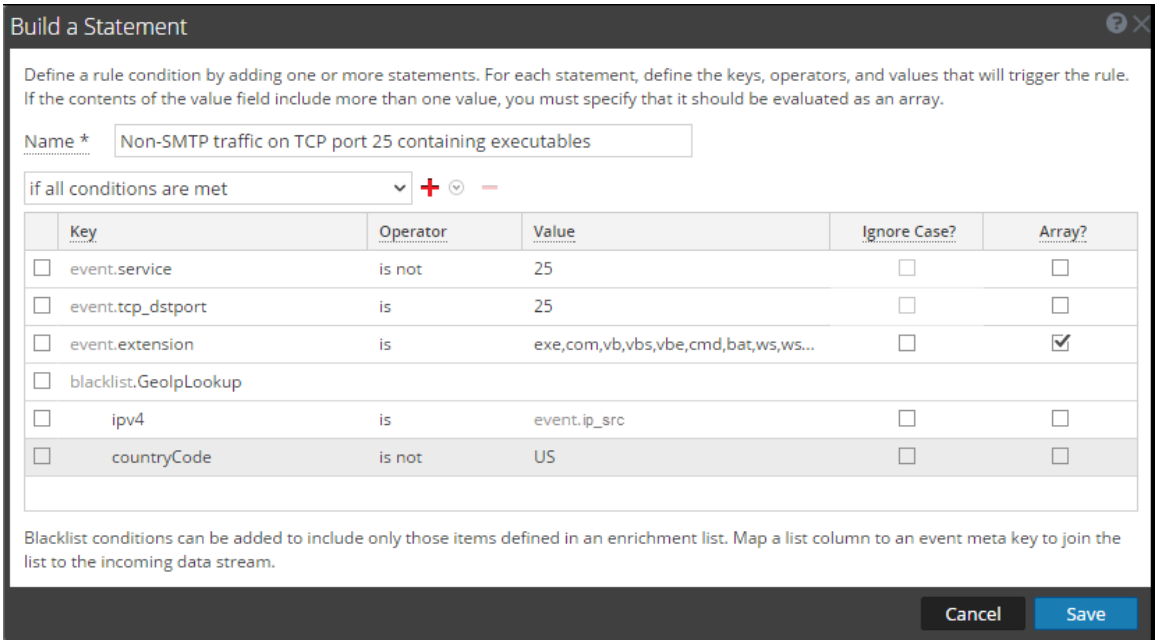
To Add a Blacklist

You use a blacklist to ensure that specified events trigger the rule. Blacklists can be based on geographic location or by customer-defined enrichment CSV sources. For example, you can specify that the rule only includes results from Germany.

1. After you add a meta condition, click + and select **Add Blacklist Condition**.
2. In the **Enter Blacklist Name** field, select an enrichment source. Any enrichment source loaded from a CSV or a named window in Esper can be used as source for a blacklist.
3. If you used a GeoIP source for the blacklist, ipv4 is automatically entered for the subcondition. Enter the meta value for the corresponding value field. For example, enter ipv4 is ip_src to ensure the GeoIP records are selected based upon the ip_src being found in the GeoIP lookup database. In addition, if you used a GeoIP source for the blacklist, you might want to add a subcondition to specify the geographic region to include in the rule results. For example, to specify that the rule only includes results for Germany, enter "*CountryCode is DE*".

Example: Blacklist

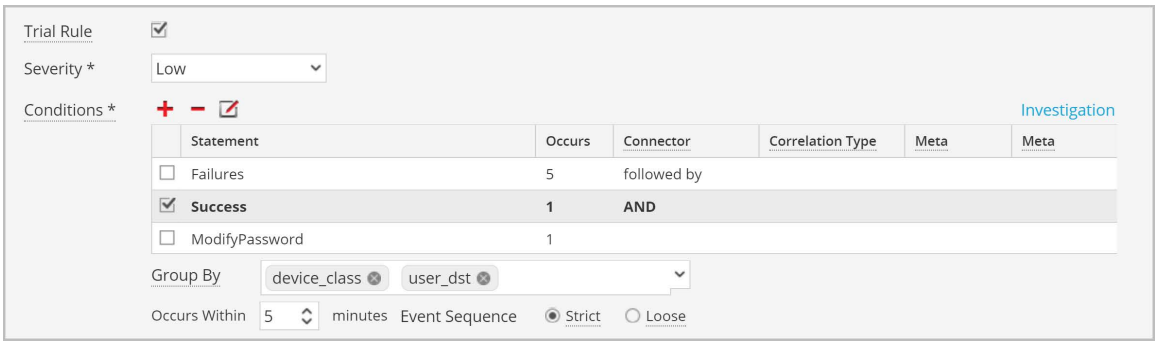
The following statement shows a blacklist statement for a rule that monitors for non-SMTP traffic on TCP destination port 25 containing an executable from countries that are outside of the United States.



Statement	Description
service is not 25	The traffic is not SMTP traffic.
tcp_dstport is 25	The traffic is running on TCP port 25.
extension is exe, com,vb,vbs,vbe,cmd,bat,ws,wsf,src,sh	The file extension is an executable.
GeoIpLookup	The blacklist is based on a GeoIPLookup source.
ipv4 is ip_src	The GeoIP records are selected based upon the ip_src being found in the GeoIP lookup database.
countryCode is not US	When looking up the IP address Event.ip_src in the GeoIP database, the record it returns does not contain "US" in the countryCode field.

Example: Ignoring Case, Strict Pattern Matching, and Using The *Is Not Null* Operator

The following example uses the ability to ignore case, exclude null values, and create a strict pattern match to ensure that it returns the expected rule results. The following conditions make up the rule:

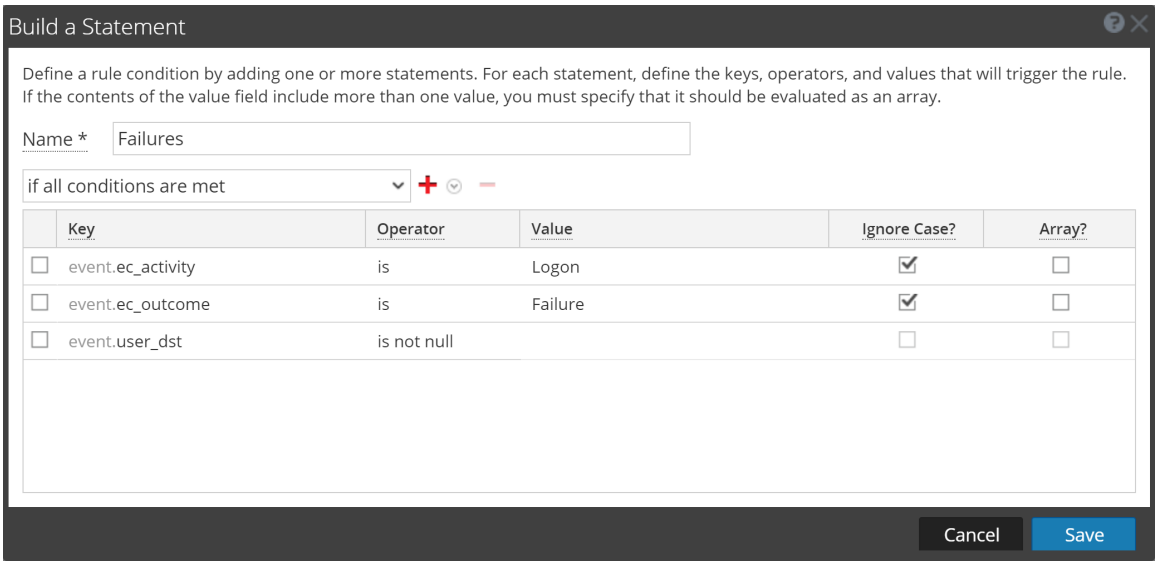


Rule Condition	Description
Failures	This condition searches for five failed logins with a "followed by" connector, meaning that the condition (Failures) must be followed by the next condition (Success).

Rule Condition	Description
Success	This condition searches for one successful login.
ModifyPassword	This condition searches for an instance where the password is modified.
GroupBy: user_dst, device class	The GroupBy field ensures that all the previous conditions are grouped by the user_dst meta (the user destination account) and device class. This is important to the construction of the rule because the rule attempts to find a case where a user has attempted to log into the same destination account multiple times, finally logged in successfully, and then changed the password. Grouping by device class ensures that the user logged in from the same machine attempted to log into an account multiple times. The rule may give unexpected results if you do not group the results.
Occurs within 5 minutes	The time window for the events to occur is five minutes. If the events occur outside of this time window, the rule does not trigger.
Event Sequence: Strict	The event sequence is configured for a strict pattern match. This means that the pattern must match exactly as it is specifiedwith no intervening events. Strict pattern matching allows you to ensure that the Esper engine only generates alerts for rules that exactly match the pattern you want to find. For example, a common rule might be to search for five failed logins followed by a successful login. If you select a loose pattern match, this rule will trigger if there are any number of successful logins between the failed logins. Since the point of the rule is to find frequent *and* sequential login attempts, a strict match is required to ensure that you get the results you expect.

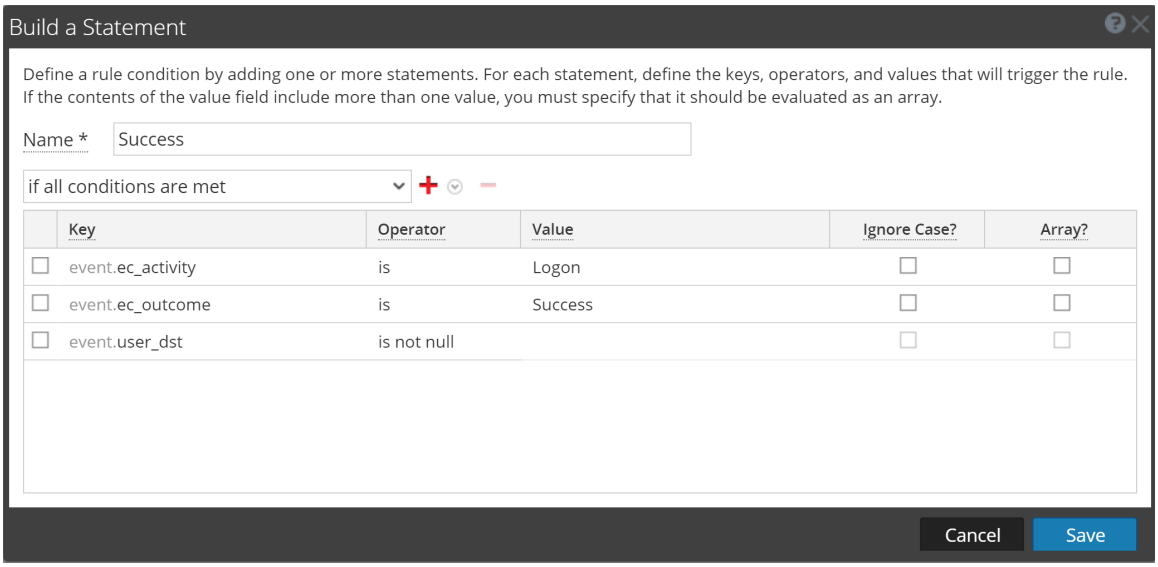
Note: Each of these conditions is explained in further detail in the sections below.

For each condition, a statement is built in the Rule Builder. The following statement makes up the Failures condition:



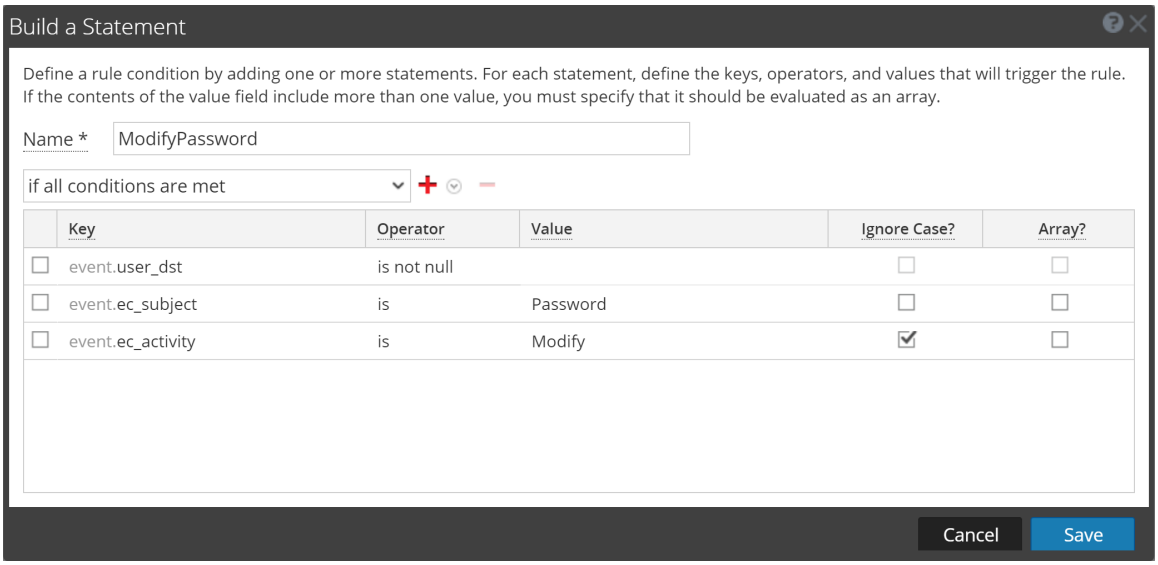
Rule Statement	Description
ec-activity is Logon (ignore case)	Identifies activity that attempts to log on to a system. The **Ignore Case** field is designed for use with string and string array values. By choosing the **Ignore Case** field, the query will treat all string text as a lowercase value. You may want to use this field if you are unsure what case may be used when logging a particular event. Because the case is ignored, the rule can trigger if the activity is logged as Logon, logon, or LoGoN.
ec_outcome is Failure (ignore case)	Identifies activity outcome logged as "failure." Because the case is ignored, the rule can trigger if the activity is logged as "failure", "Failure," or "FaiLuRe."
user_dst is not null	Ensures that the condition is only true if user_dst is populated. The **is not null** operator allows you to ensure that a field returns a value. You may want to use this field when a rule depends on a particular field returning a value. For example, you want to create a rule that identifies the same user attempting to log into the same destination account multiple times (potentially a password-guessing attack). If the field that represents the user destination account is empty, you don't want the rule to trigger. To ensure the field contains a value, you use the **is not null** operator.

The following statement makes up the Success condition:



Rule Statement	Description
ec_activity is Logon	Identifies logon activity.
ec_outcome is Success	Identifies a logon that is successful.
user_dst is not null	Ensures that user destination account field must be populated for the condition to be true.

The following statement makes up the ModifyPassword condition:

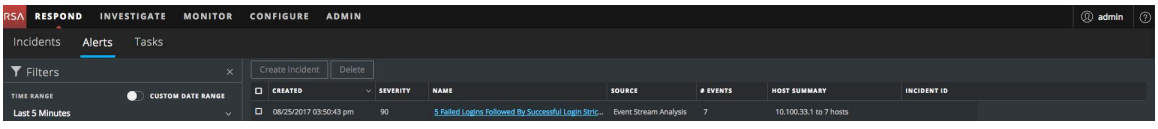


Rule Statement	Description
user_dst is not null	Ensures the user destination account field must be populated for the condition to be true.
ec_subject is Password	Identifies a subject of Password.
ec_activity is Modify	Identifies activity where the password was modified.

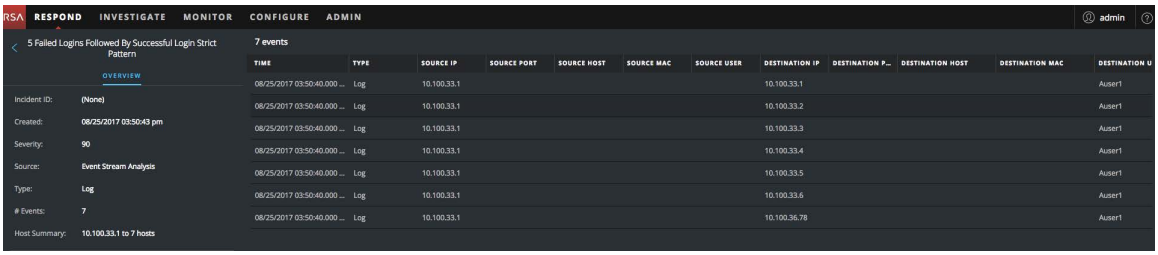
Example Results

When the alert fires for the example rule, you can see that the rule triggered for seven events, and that each event contains a user. You can also see that the events follow a strict pattern: five failed login events, followed by a successful login event, followed by a modification to the account.

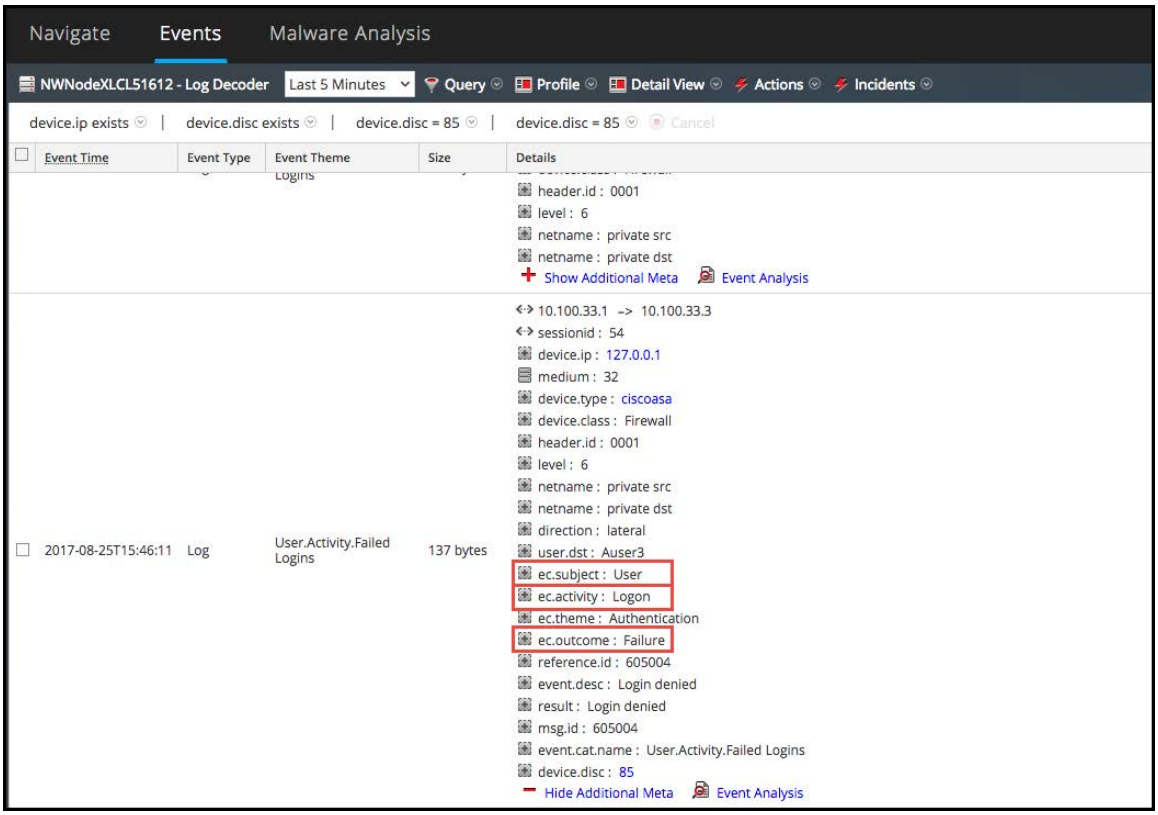
The following figure shows the alert in the Respond Alerts List view.



The next figure shows the events in the alert in the Respond Alert Details view.



Drilling down into the Investigation module by clicking on the source for one of the events, you can see the case for each of the string values. Because you used **Ignore Case**, the rule would trigger if the string values were upper or lower case.

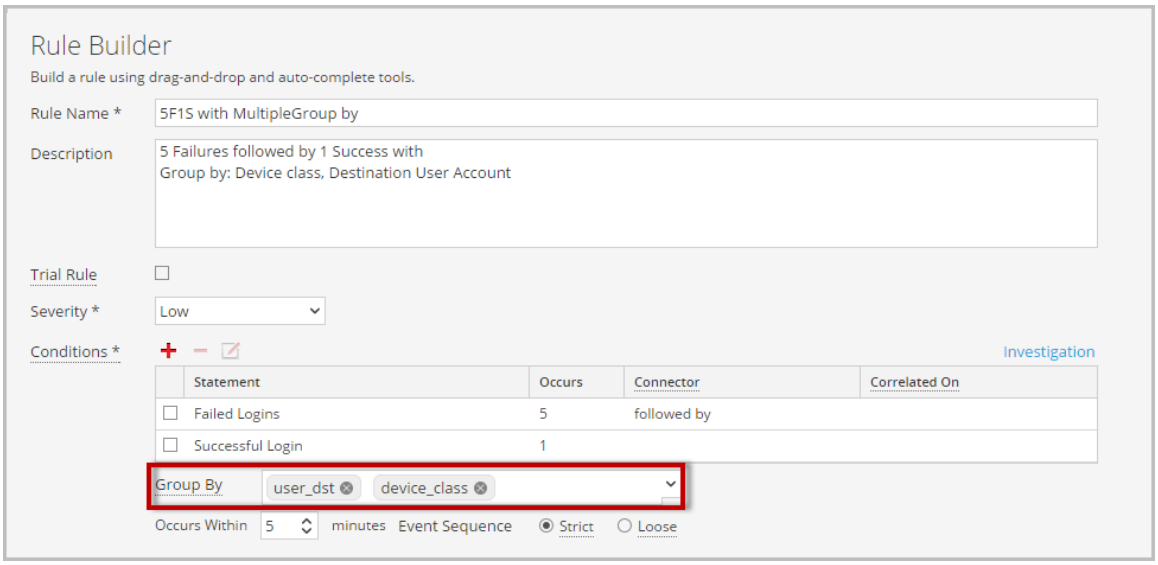


Example: Grouping the Rule Results

The **Group By** field allows you to group and filter rule results. For example, suppose that there are three user accounts; Joe, Jane, and John and you use the **Group By** meta, user_dst. The result will show events grouped under the accounts for Joe, Jane, and John.

You can also group by multiple keys, which can further filter rule results. For example, you might want to group by user destination account and machine to see if a user logged into the same destination account from the same machine attempts to log into an account multiple times. To do this, you might group by device_class and user_dst.

The following example shows a rule grouped by device_class and user_dst.

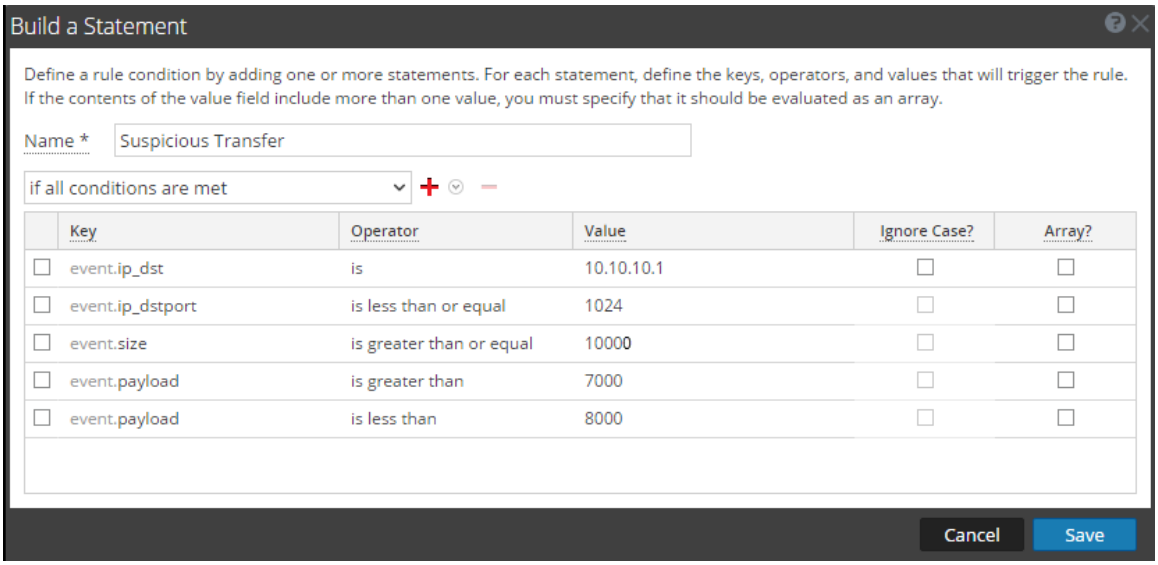


Rule Condition	Description
Failed Logins	Identifies five failed login attempts (must be followed by the next condition; i.e., the five failed logins must be followed by a successful login).
Successful Login	Identifies one successful login.
Group By: user_dst and device_class	Groups the rule results by user_dst (user destination account) and device_class (type of machine the user is logging in from). This allows the rule to look for a user logged in from the same machine to the same destination account, resulting in a much more targeted rule result.
Occurs within 5 minutes with a strict pattern match	The events must occur within five minutes, and the pattern matching is strict, meaning it must follow the pattern exactly for the rule to trigger.

Example: Working with Numeric Operators

Numeric operators allow you to write rules against numeric values, such as specifying that a value is greater than, less than, or equal to a specific value. This is useful particularly for cases where you might want to specify a numeric threshold, i.e., *payload is greater than 7000*.

The following example attempts to identify a data transfer to a particular destination through the common ports where the transfer size is high and the payload is in a suspicious range.



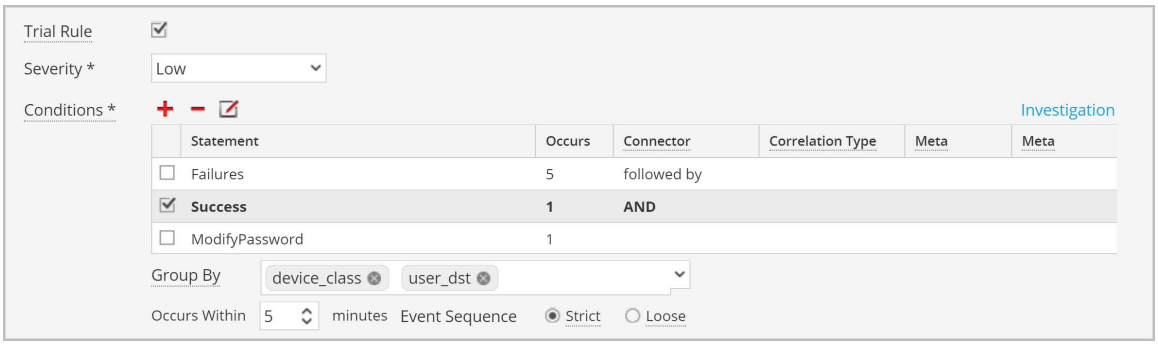
Rule Statement	Description
ip_dst is 10.10.10.1	The destination port is 10.10.10.1.
ip_dstport is greater than or equal to 1024	The destination port is in a commonly used port range, 1024 or greater.
size is greater than or equal to 10000	The size of the transfer is 10000 or greater, which is a suspiciously large transfer.
payload is greater than 7000	The payload is between 7000 and 8000, which is a suspiciously large payload.
payload is less than 8000	The payload is between 7000 and 8000, which is a suspiciously large payload.

Step 3. Add Conditions to a Rule Statement

This topic provides instructions to add conditions, such as specifying a certain time frame, to a rule statement. When you build a statement, you specify what a rule detects. You add conditions to make further stipulations, such as how many times or when the criteria must occur.

Example

The following graphic shows an example of the conditions for Rule Builder statements. Combined, the statements and conditions comprise the rule criteria.



This rule detects 5 failed logon attempts followed by one successful logon, which could be the sign that someone has hacked into user account. This is the criteria for the rule:

A. 5 failed logons are required.

B. 1 successful logon must follow the failures

C. A password was changed.

D. All events must occur within 5 minutes.

E. Group alerts by user (user_dst), because steps A and B must be performed on the same user destination account. Also, group by machine (device_class) to ensure that the user logged in from the same machine attempts to log into an account multiple times.

F. The match is a strict pattern, meaning that the pattern must match exactly with no intervening events.

Procedure

To add conditions to a rule statement:

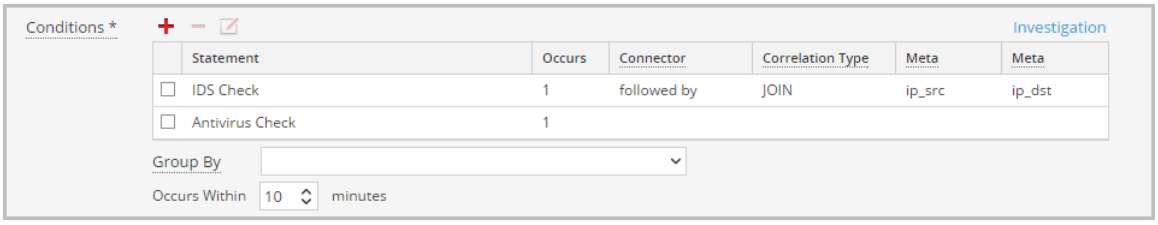
1. In the **Conditions** section, select a statement and click  .
2. For **Occurs**, enter a value to specify how many occurrences are required to meet the rule criteria.

3. If you have multiple statements, in the **Connector** field select a logical operator to join one statement to another:
   - followed by
   - not followed by
   - AND
   - OR
4. **Correlation Type** applies only to **followed by** and **not followed by**. If you choose a correlation type of SAME, select one meta to correlate on, and if you choose a correlation type of JOIN, select two meta to correlate on. You may want to use JOIN if you are trying to correlate on meta from two different data sources. For example, say you want to correlate an AV alert with an IDS alert. See the examples below for a use case where two meta from different sources are joined.
5. If events must happen within a specific timeframe, enter a number of minutes in the **Occurs Within** field.
6. Choose whether the pattern must follow a **Strict** match or a **Loose** match. If you specify a strict match, this means that the pattern must occur in the exact sequence you specified with no additional events occurring in between. For example, if the sequence specifies five failed logins (F) followed by a successful login (S), this pattern will only match if the user executes the following sequence: F,F,F,F,F,S. If you specify a loose match, this means that other events may occur within the sequence, but the rule will still trigger if all of the specified events also occur. For example, five failed login attempts (F), followed by any number of intervening successful login attempts (S), followed by a successful login attempt might create the following pattern: F,S,F,S,F,S,F,S,F,S which would trigger the rule despite the intervening successful logins.
7. Choose the fields to group by from the dropdown list. The **Group by** field allows you to group and evaluate the incoming events. For example, in the rule that detects 5 failed logon attempts followed by 1 successful attempt, the user must be the same, so user_dst is the **Group By** meta key. You can also group by multiple keys. Using the previous example, you might want to group by user and machine to ensure that the same user logged in from the same machine attempts to log into an account multiple times. To do this, you might group by device_class and user_dst.

Example

The following graphic shows an example of the conditions for a rule that allow you to evaluate the same entities across multiple devices so you can accomplish complex use cases. For example, you can create a rule that triggers if an IDS (Intrusion Detection System) alert is followed by an AV(Anti-virus) alert for the same workstation. The work station key is not the same between the two (IDS & AV) sources, so you can perform a JOIN in order to evaluate the different entities.

In the IDS alert, the workstation is identified by the source IP address from the IDS alert, and would be compared to the destination IP address from the AV alert.



This is the criteria for the rule:

A. An IDS alert occurs.

B. The destination IP address from the AV alert and source IP address for the workstation from the IDS alert are joined to allow you to view the same entities across different sources.

C. An Antivirus alert follows the IDS alert.

Add an Advanced EPL Rule

This topic provides instructions to define rule criteria by writing an EPL query. EPL is a declarative language for handling high-frequency time-based event data. It is used to express filtering, aggregation, and joins over possibly sliding windows of multiple event streams. EPL also includes pattern semantics to express complex temporal causality among events.

Write an advanced EPL rule when rule criteria is more complex than what you can specify in Rule Builder.

It is outside the scope of this guide to explain EPL syntax.

- For EPL Documentation, see http://www.espertech.com/esper/documentation.php.
- For the EPL Online Tool, see http://esper-epl-tryout.appspot.com/epltryout/mainform.htm

Prerequisites

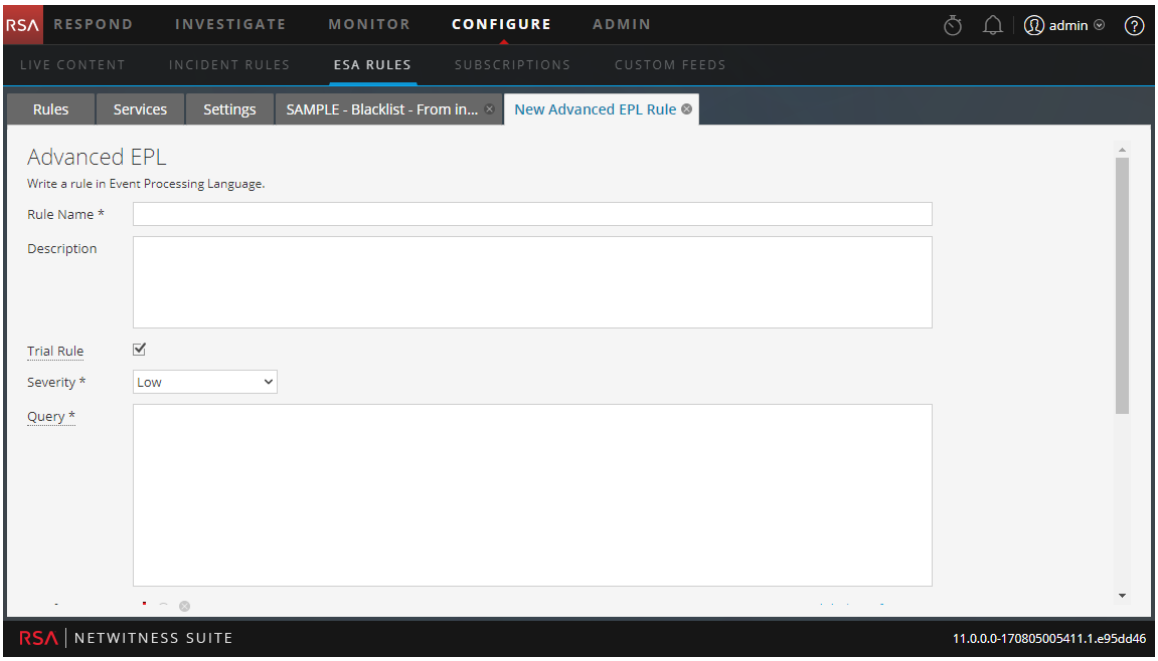
The following are prerequisites for adding an advanced rule:

- You must know Event Processing Language (EPL).
- You must understand ESA Annotations to mark which EPL statements are linked to generating alerts.

Procedure

To add an Advanced EPL rule:

1. Go to **CONFIGURE > ESA Rules**.
2. In the **Rule Library**, select + > **Advanced EPL**.



3. Type a unique, descriptive name in the **Rule Name** field.

   This name will appear in the Rule Library so be specific enough to distinguish the rule from others.

4. In the **Description** field, explain which events the rule detects.

   The beginning of this description will appear in the Rule Library

5. Select **Trial Rule** to automatically disable the rule if all trial rules collectively exceed the memory threshold.

   Use trial rule mode as a safeguard to see if a rule runs efficiently and to prevent downtime caused by running out of memory. For more information, see Work with Trial Rules.

6. For **Severity**, classify the rule as Low, Medium, High or Critical.
7. To define rule criteria, write a **Query** in EPL.

Note: For all meta key names, use an underscore not a period. For example, `ec_outcome` is correct but `ec.outcome` is not.

8. For dynamic statement name generation in ESA, you must enclose the meta keys in curly brackets and include this annotation in the syntax:

   ```
   @Name("RIG {ip_src} {alias_host} {ec_activity}")
   ```

   where,

   - RIG is the static part of the statement name
   - {ip_src}, {alias_host}, {ec_activity} is the dynamic part of the statement name

   Note: If any of the metas in the dynamic part of the statement name has a null value, it is displayed as a static text.

   If a rule should generate an alert, include this ESA annotation in the syntax:

   ```
   @RSAAlert
   ```

   For more information on ESA Annotations, see ESA Annotations.

Event Processing Language (EPL)

This topic describes Event Processing Language (EPL), a declarative language for dealing with high frequency time-based event data. ESA uses Event Processing Language (EPL), a declarative language for dealing with high frequency time-based event data. It is used for express filtering, aggregation, and joins over possibly sliding windows of multiple event streams. EPL also includes pattern semantics to express complex temporal causality among events. It can perform, but is not limited to, the following functions:

- Filter Event
- Alert Suppression
- Compute percentages or rations
- Average, count, min and max for a given time window
- Correlate events arriving in multiple stream
- Correlate events that arrive out of order
- On-Off Windows
- Followed-by and Not Followed-by support
- Regex filter support

Databases require explicit querying to return meaningful data and are not suited to push data as it changes. The developer must implement the temporal and aggregation logic himself. By contrast, the EPL engine provides a higher abstraction and intelligence and can be thought of as a database turned upside-down. Instead of storing the data and running queries against stored data, EPL allows applications to store queries and continuously run the data through. Response from the EPL engine is real-time when conditions occur that match user defined queries.

Advanced ESA rules require correct character case, but in the Investigation view all characters are converted to lowercase. However, the meta may not be lowercase despite appearances in the Investigation view. To ensure you are using the correct case, RSA recommends you use the *toLowerCase()* function. For example,

```
@RSAAlert(oneInSeconds=0)
SELECT * FROM Event(
/* Statement: Download PDF File */
(filetype.toLowerCase() IN ( 'pdf' ) AND medium IN ( 1 ))
OR
/* Statement: Download EXE File */
(filetype.toLowerCase() IN ( 'windows_executable' , 'x86 pe' , 'windows executable' ) AND medium IN ( 1 ))
).win:time(5 Minutes)
MATCH_RECOGNIZE (
PARTITION BY ip_src
MEASURES E1 as e1_data , E2 as e2_data
PATTERN (E1+ E2)
DEFINE
E1 as (E1.filetype.toLowerCase() IN ( 'pdf' ) AND E1.medium IN ( 1 )),
E2 as (E2.filetype.toLowerCase() IN ( 'windows_executable' , 'x86 pe' , 'windows executable' ) AND E2.medium IN ( 1 ))
```

For the purposes of online help, basic statements are used to illustrate how to set up ESA; however, for more information about writing EPL statements, the http://www.espertech.com site provides tutorials and examples.

Note: ESA supports Esper version 5.3.0.

ESA Annotations

This topic describes annotations that NetWitness Suite provides to use in advanced EPL rules.

@RSAAlert Annotation

The @RSAAlert annotation is used to mark which EPL statements are linked to generating alert notifications. It is designed to work with the alert notification suppression feature in the Rule Builder user interface.

The @RSAAlert annotation can be useful when working with alert notifications, especially if you want to filter notifications, such as sending one notification for each user that triggers an alert.

For example, suppose you want to generate alert notifications for login failures. You could add the following statement:

*@RSAAlert select * from event(msg_id="login_fail")*

Event number	Message ID	username	src_IP	Time
1	login_fail	alice	1.2.3.4	10:00
2	login_fail	alice	1.2.3.4	10:01
3	login_fail	alice	6.7.8.9	10:01
4	login_fail	bob	1.2.3.4	10:01
5	login_fail	alice	1.2.3.4	10:03

For the above statement, five alert notifications are generated.

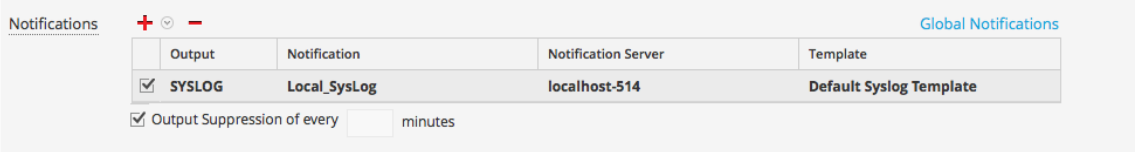
However, suppose you wanted to modify the statement to generate one alert for each separate username. You can use the *identifier* attribute. For example, the statement *@RSAAlert (identifier={"username"}) SELECT* FROM Event(msg_id="login_fail")* generates one notification for the first alert for "bob" and one for the first alert for "alice." Subsequent alerts for "bob" and "alice" are ignored.

You can further distinguish the users by adding details via the identifier variable. For example, you can distinguish by user and IP address using the following statement: *@RSAAlert(identifier={"username", "src_ip"}) SELECT* FROM Event(msg_id="login_fail")*. Then, you would see notifications generated by user name and IP address (one alert for "alice" at 1.2.3.4, another alert for "alice" at 6.7.8.9, and an alert for "bob" at 1.2.3.4).

To Use Identifiers with Alert Notification Suppression:

The @RSAAlert annotation is designed to work with the alert notification suppression feature in the Rule Builder user interface. To do this:

1. Create a rule in the Rule Builder user interface, and select the alert suppression feature when configuring notifications.



2. Copy the code from the Rule Builder rule into a new advanced rule.
3. Configure the advanced rule to include identifiers (as described above) and save the advanced rule.
4. Delete the original rule builder rule.

@RSAPersist Annotation

The @RSAPersist annotation is used to mark a named window as an ESA managed window for persistence. By marking the named window as an ESA managed window, ESA periodically writes the contents of the window to disk and restores them back if the window is un-deployed and re-deployed. The systems take a snapshot just before the module is un-deployed and the window is removed. Conversely, it restores the window contents from the snapshot just after the module is re-deployed. This ensures that the contents of the window are not lost if the module state is altered or if the ESA service goes down.

For example, consider a named window, `DHCPTracker` that holds a mapping from IP addresses to each assigned hostname.You can annotate the statement with the @RSAPersist annotation as:

```
@RSAPersist
 create window DHCPTracker.std:unique(ip_src) as (ip_src string, alias_
host string);
 insert into DHCPTracker select IP as ip_src, HostName as alias_
host from DHCPAssignment(ID=32);
```

Note: All windows definitions are not suitable for persistence. `@RSAPersist` annotation must be used with care. If the window has timed-records or if it depends on time based constraints it is very likely that the reverted snapshots will not restore it to the correct state. Also, any changes to the window definition will invalidate the snapshots and reset the window to a blank state. The system does not do any semantic analysis to determine if the changes to the window definition are conflicting or not. Note that other parts of a module (i.e. other than the particular CREATE WINDOW call that defines the window) may change, without invalidating the snapshots.

@UsesEnrichment (10.6.1.1 and later)

The @UsesEnrichment can be used in advanced EPL rules to reference enrichments. In order to synchronize enrichments with ESA, all enrichment dependencies in EPL rules must be referenced with the @UsesEnrichment annotation.

The @UsesEnrichment annotation uses the following format:

```
@UsesEnrichment(name= '<enrichment_name>')
```

For example, the following EPL references a whitelist enrichment:

```
@UsesEnrichment(name = 'Whitelist')
@RSAAlert
SELECT * FROM Event(ip_src NOT IN (SELECT ip_address FROM Whitelist))
```

@Name

The @Name is the statement name defined in ESA advanced rules. It is used to dynamically generate statement names in ESA alerts. The statement name of only an alert triggering statement is displayed. This annotation has meta keys enclosed in curly brackets.

The @Name annotation uses the following format:

```
@Name("<static_part_of_statement_name> {meta_key1} {meta_
key2}…")
```

For example, the following EPL references meta keys *ip_src* and *user_name* whose values will be dynamically generated.

@Name(“Login Event to {ip_src} by {user_name}”)

Note: You can specify any number of meta keys in the statement for dynamic statement name generation.
The length of individual meta key is limited to 64, after which the value is truncated and appended with “…”.
The length of the dynamic generation of statement name is limited to 128, after which the value is truncated to 128 and appended with “…”. All the remaining values post truncation will be treated as static values.

Sample Advanced EPL Rules

Following are the examples of Advanced ESA rules. Each example has multiple ways of implementing the same use-case.

Example #1:

Create an user account and delete the same user account in 300s. User information is stored in user_src meta.

EPL #1:

Rule Name	CreateuseraccountFollowedbyDeletionof Useraccount1
Rule	Create a user account followed by an action to delete the same user account in

Description	300 seconds.
Rule Code	SELECT * FROM Event(ec_subject='User' AND ec_outcome='Success' AND user_src is NOT NULL AND ec_activity IN ('Create', 'Delete')).win:time(300 seconds) match_recognize (partition by user_src measures C as c, D as d pattern (C D) define C as C.ec_activity='Create' , D as D.ec_activity='Delete');
Note	• Filter events needed for pattern in given time frame. Filter conditions should be such that only required events are passed to match recognize function. In this case, they are create and delete user account Events. i.e. Event(ec_subject='User' AND ec_outcome='Success' AND user_src is NOT NULL AND ec_activity IN ('Create', 'Delete') • Partition by creates buckets. In this case, esper creates buckets per value of user_src. And hence value of user_src is common between both events. • Define pattern you want. Right now it is set to Create Followed by Delete. You can do multiple creates followed by delete (C+ D). Pattern is very similar to regular expression. • Most efficient use case.

EPL #2:

Rule Name	CreateuseraccountFollowedbyDeletionof Useraccount2
Rule Description	Create a user account followed by an action to delete the same user account in 300 seconds.
Rule Code	SELECT * from pattern[every (a= Event(ec_subject='User' AND ec_outcome='Success' AND user_dst is NOT NULL AND ec_activity IN ('Create')) ->(Event(ec_subject='User' AND ec_outcome='Success' AND user_dst is NOT NULL AND ec_activity IN ('Create') AND

	`user_src = a.user_src)` `) )where timer:within(300 Sec) ];`
Note	• Lets say same user is created twice and deleted once in that order. Then the above pattern will fire 2 alerts. • A thread is created for every User creation. • There is no way to control threads. It is important to have time bonds and preferably small intervals.

Example #2:

Detect pattern where user created followed by login by same user and user is deleted in end. In case of windows logs user info is stored in either user_dst or user_src depending on event.

user_src(create) = user_dst(Login) = user_src(Delete)

EPL #3:

Rule Name	CreateUserLoginandDeleteUser
Rule Description	Detect a pattern where a user creates a User account followed by login by the same user followed by deletion of the User account.

Rule Code	`SELECT * FROM Event(ec_subject='User'`
	`and ec_activity in ('Create','Logon','Delete')`
	`and ec_theme in ('UserGroup', 'Authentication')`
	`and ec_outcome='Success'`
	`).win:time(300 seconds)`
	`match_recognize (measures C as c, L as l, D as d`
	`pattern (C L D)`
	`define`
	`C as C.ec_activity = 'Create',`
	`L as L.ec_activity = 'Logon' AND L.user_dst = C.user_src,`
	`D as D.ec_activity = 'Delete' AND D.user_src = C.user_src`
	`);`

Note	• Since user_src/user_dst is not common across all events we can't use partition. It will be 1 single bucket running 1 pattern at a time. For example, for user 1 and 2 if the stream of events are C1C2L1D1, C1L1C2D1, there will be no alert because C1 thread got reset by C2. Alert will be fired only if C1L1D1 are in order and no other event either from same user or other user falls in between.

	• Another solution would be to use Named Window and merge user_dst and user_src into single column and then run match recognize. (EPL #3). • Pattern can also be used. You might get more alerts than expected. (EPL #4).

EPL #4: Using NamedWindows and match recognize

Rule Name	CreateUserLoginandDeleteUser
Rule Description	Detect a pattern where a user creates a User account followed by login by the same user followed by deletion of the User account.

Rule Code	`@Name('NormalizedWindow')create window FilteredEvents.win:time(300 sec) (user String, ecactivity string, sessionid Long);`
	`@Name('UsersrcEvents') Insert into FilteredEvents select user_src as user, ec_activity as ecactivity, sessionid from Event(ec_subject='User' and ec_activity in ('Create','Delete') and ec_theme in ('UserGroup', 'Authentication') and ec_outcome='Success' and user_src is not null );`
	`@Name('UsrdstEvents') Insert into FilteredEvents select user_dst as user, ec_activity`
	`as ecactivity, sessionid from Event(ec_subject='User' and ec_activity in (Logon') and ec_theme in ('UserGroup', 'Authentication') and ec_outcome='Success' and user_dst is not null );`
	`@Name('Pattern')`
	`@RSAAlert(oneInSeconds=0, identifiers={"user"})`
	`select * from FilteredEvents`
	`match_recognize (`
	`partition by user`
	`measures C as c, L as l, D as d`
	`pattern (C L+D)`
	`define C as C.ecactivity= 'Create',`
	`L as L.ecactivity= 'Logon',`
	`D as D.ecactivity='Delete'`
	`);`

EPL #5: Using Every @RSAAlert(oneInSeconds=0, identifiers={"user_src"})

SELECT a.time as time,a.ip_src as ip_src,a.user_dst as user_dst,a.ip_dst as ip_dst,a.alias_host as alias_host from pattern[every (a=Event (ec_subject='User' and ec_activity='Create' and ec_theme='UserGroup' and ec_outcome='Success') -> (Event(ec_subject='User' and ec_activity='Logon' and ec_theme='Authentication' and user_src=a.user_dst) -> b=Event (ec_subject='User' and ec_activity='Delete' and ec_theme='UserGroup' and user_dst=a.user_dst))) where timer:within(300 sec)];

Rule Name	CreateUserLoginandDeleteUser
Rule Description	Detect a pattern where a user creates a User account followed by login by the same user followed by deletion of the User account.

Example #3:

Excessive login failures from same sourceIP

EPL #6: @RSAAlert(oneInSeconds=0, identifiers={"ip_src"})

Rule Name	ExcessLoginFailure
Rule Description	The same user tried logging in from the same Source IP and faced login failures
Rule Code	`SELECT * FROM Event( ip_src IS NOT NULL AND ec_activity='Logon' AND ec_outcome = 'Failure' ).std:groupwin(ip_src).win:time_length_batch(300 sec, 10) GROUP BY ip_src HAVING COUNT(*) = 10;`
Note	• Creates window per ip_src • Uses time_length_batch: Looks at events in batches(tumbling window). Every event will be part of only 1 window. Window releases events either when time elapses or count is reached. • One of issues with tumbling windows that events occurring towards end of batch might not lead to an alert. In below sequence of events at t=301 even though 10 login failures occurred for same login in last 300 secs there will be no alert because batch of events was dropped at t=300

	Time t Login Failures for Specific Users Alert Time Batch 0 0 0 1 295 6 0 1 299 3 0 1 301 1 0 2 420 6 0 2 550 3 0 2 600 0 0 3 720 6 0 3 850 3 0 3 900 1 1 3 ends and 4 begins • Above problem can be resolved using win:time windows (EPL#7)instead of win:time_length_batch windows. • Outer group by is to control events when time elapses. Say you have 9 events at end of 60 secs, esper engine will push those 9 events to listener. Group by and count will restrict it since count is not equal to 10. • Time and count can be modified as needed.

EPL #7: @RSAAlert(oneInSeconds=0, identifiers={"ip_src"})

Rule Name	ExcessLoginFailure
Rule Description	The same user tried logging in from the same Source IP and faced login failures

Rule Code	`SELECT * FROM`
	`  Event(`
	`  ip_src IS NOT NULL AND ec_activity='Logon' AND ec_outcome = 'Failure'`
	`  ).std:groupwin(ip_src).win:time (300 sec) GROUP BY ip_src HAVING COUNT(*) = 10`

Note	• This is sliding window and hence once alert is fired for a set of events they can be used for another alert as well till time has passed. • If 10 events were involved in causing alert only last event will appear

	• If < or > are used then you might see more than 1 alert. You should use alert suppression accordingly.

Example #4:

Multiple failed logins from multiple different users from same source to same destination, a single user from multiple different sources to same destination.

EPL #8: using groupwin , time_length_batch and unique

Rule Name	MultiplefailedLogins
Rule Description	There are multiple failed logins for the following cases: - From multiple users from same source to same destination. - Single user from multiple sources to the same destination.
Rule Code	`SELECT * FROM Event( ec_activity='Logon' AND ec_outcome='Failure' AND ip_src IS NOT NULL AND ip_dst IS NOT NULL AND user_dst IS NOT NULL ).std:groupwin(ip_src,ip_dst).win:time_length_batch(300 seconds, 5}).std:unique(user_dst) group by ip_src,ip_dst having count(*) = 5;`
Note	• ip.dst and ip.src are common across all events. • user_dst is unique for all events. • Alert is fired when there are atleast 5 different users try to login from same ip.src and ip.dst combination.

Example #5:

No Log traffic from a device in a given timeframe.

EPL #9: using groupwin , time_length_batch and unique

Rule Name	NoLogTraffic
Rule Description	There is no log traffic observed from a device in a given time frame.
Rule Code	`SELECT * FROM pattern [every a = Event(device_ip IN ('10.0.0.0','10.0.0.1') AND medium = 32) -> (timer:interval (3600 seconds) AND NOT Event(device_ip = a.device_ip AND`

Note	`device_type = a.device_type AND medium = 32))];`
	• Rule only detects sudden loss of traffic. It won't alert if there is no traffic to begin with. You need at least 1 event for rule to alert. • List of device ip address or device hostnames as input. Only these systems will be tracked. • Time input is required. Alert is fired when time interval between events exceeds input time.

Example #6:

Multiple Failed Logins NOT followed by a Lockout event by the same user.

EPL #10: using groupwin , time_length_batch and unique

Rule Name	FailedloginswoLockout
Rule Description	There are multiple failed logins that are not followed by Lockout event by the same user.

Rule Code	`SELECT * FROM pattern [every-distinct(a.user_dst, a.device_ip, 1 msec) (a= Event(ec_activity='Logon' and ec_outcome='Failure' and user_dst IS NOT NULL)-> [2]( Event( device_ip =a.device_ip and ec_activity='Logon' and ec_outcome='Failure' and user_dst=a.user_dst)`
	`AND NOT Event( ( ec_activity='Logon' and ec_outcome='Success' and device_ip = a.device_ip and user_dst=a.user_dst) or (ec_activity='Lockout' and device_ip = a.device_ip and user_dst=a.user_dst))))`
	`where timer:within(60 seconds) -> (timer:interval(30 seconds) and not Event(device_ip=a.device_ip and user_dst=a.user_dst and ec_activity='Lockout'))];`

Note	• Above query detects the absence of a Lockout Event after the occurrence of 2 failed logins from same user. • The occurrence of the multiple failed logins are timed and are assumed to occur within a certain period of time. Also, in-practice the Lockout event is assumed to occur within a short time after the occurrence of the last failed login event because the threshold value of Failed logins per user is set in a

	given domain. • In current query, every distinct will suppress new thread for combination of user and device for 1 millisec. • Time allowed for 3 failed logins is 60 secs since 1st failed attempt. Wait period for lockout event to occur is 30 secs

Example #7:

Custom functions to perform LIKE and REGEX operations for ARRAY elements.

EPL #11: @RSAAlert(oneInSeconds=0)

Rule Name	MatchLikeRegex
Rule Description	There are custom functions to perform LIKE and REGEX comparisons of array meta keys.

Rule Code	`SELECT * FROM pattern[`
	`e1=Event(matchLike(alias_host, "10.0.0.%")) AND`
	`e2=Event(matchRegex(alias_host, "10\.0\.0\.1[0-9][0-9]"))`
	`where timer:within(5 Minutes)];`

Note:
1. “.” in meta keys should be replaced with (”_”).
2. All patterns should be time bound.
3. Use of appropriate tags in front of statements
 a) @RSAPersist:
 b) @RSAAlert:

For additional details you can refer to:

- EPL Documentation: http://www.espertech.com/esper/documentation.php
- EPL Online Tool: http://esper-epl-tryout.appspot.com/epltryout/mainform.html

Working with Rules

This topic discusses additional procedures you can perform on rules. You may want to perform any of the following procedures:

Edit, Duplicate or Delete a Rule

This topic provides instructions to edit, duplicate, or delete an Event Stream Analysis (ESA) rule. When you edit a rule, ESA applies the updated criteria going forward. No changes are made to previously generated alerts.

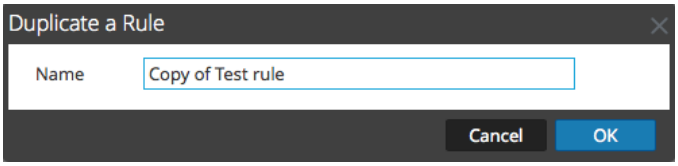
Procedures

Edit a Rule

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
   The Rules tab is displayed.
2. In the **Rule Library**, select the rule you want to edit and click .
   Depending on the rule type, the respective rule tab is displayed.
3. Modify the required parameters.
4. Click **Save**.

Duplicate a Rule

1. In the **Rule Library**, select the rule you want to duplicate and click .
2. The Duplicate a Rule dialog is displayed. The system adds **Copy of** in front of the rule name.

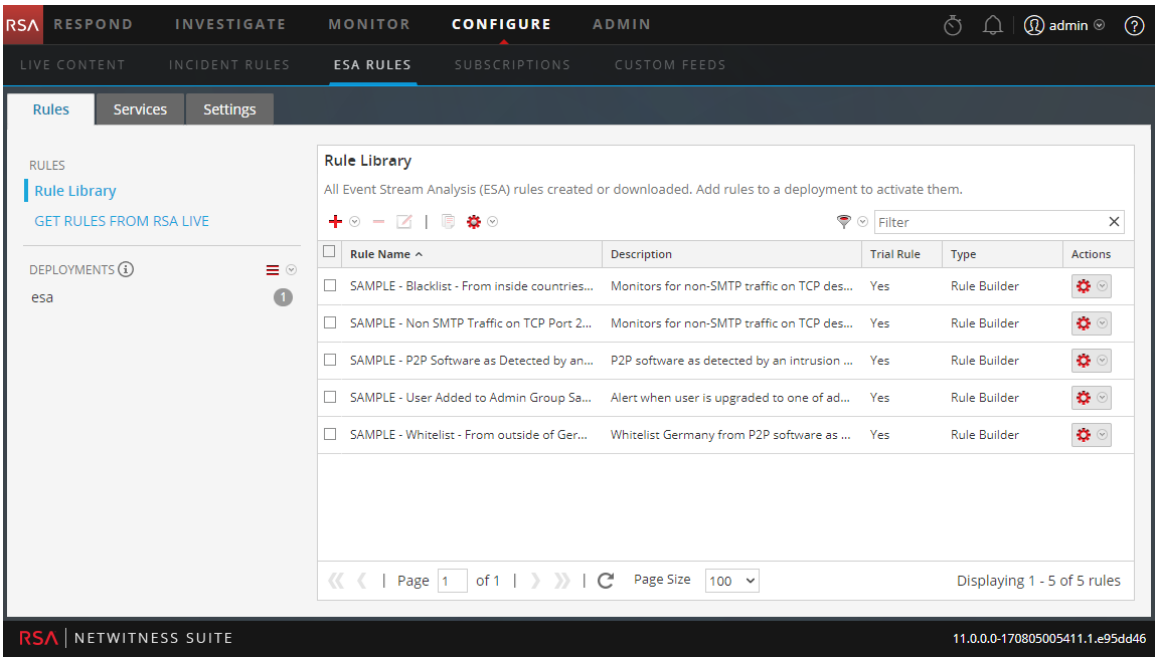


3. In the **Name** field, type a unique name for the duplicate rule and click **OK**.

A duplicate rule with the new name is added to the Rule Library.

Delete a Rule

1. Go to **CONFIGURE > ESA Rules > Rules**.
   The Rules tab is displayed.



2. In the Rule Library, select one or more rules and click .

   A warning dialog is displayed.

3. Click **Yes**.
   A confirmation message that the rule is deleted successfully is displayed and the selected rule is deleted from the Rule Library.

Filter or Search for Rules

This topic shows analysts how to specify the type of rules that display in the Rule Library.

Prerequisites

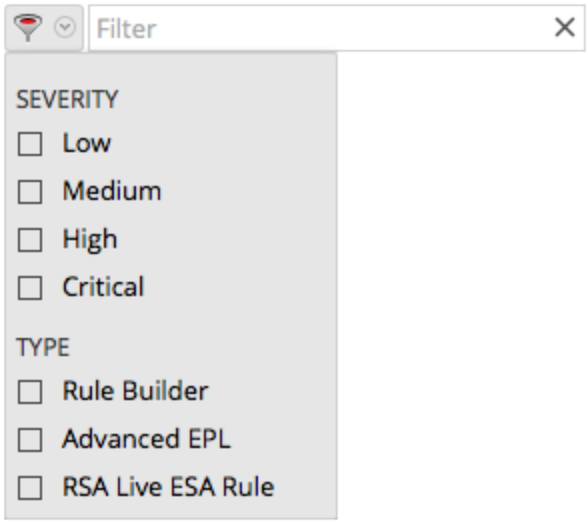
Make sure that you understand the Rule Library view components. For more information, see Rule Library Panel.

Procedures

Filter

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab is displayed by default.

2. In the **Rule Library** panel toolbar, click and select the severity and type of rules that you would like to appear in the Rule Library list. The following figure shows the Filter drop-down list.



The selected rule types appear in the list.

Search

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab is displayed by default.
2. In the **Rule Library** panel toolbar, type a rule name in the Filter field.
   The Rule Library panel lists the rules that match the names entered in the Filter field.

Import or Export Rules

The topic provides instructions to import ESA rules from a NetWitness Suite instance and to export ESA rules to your hard drive so you can keep a local copy.

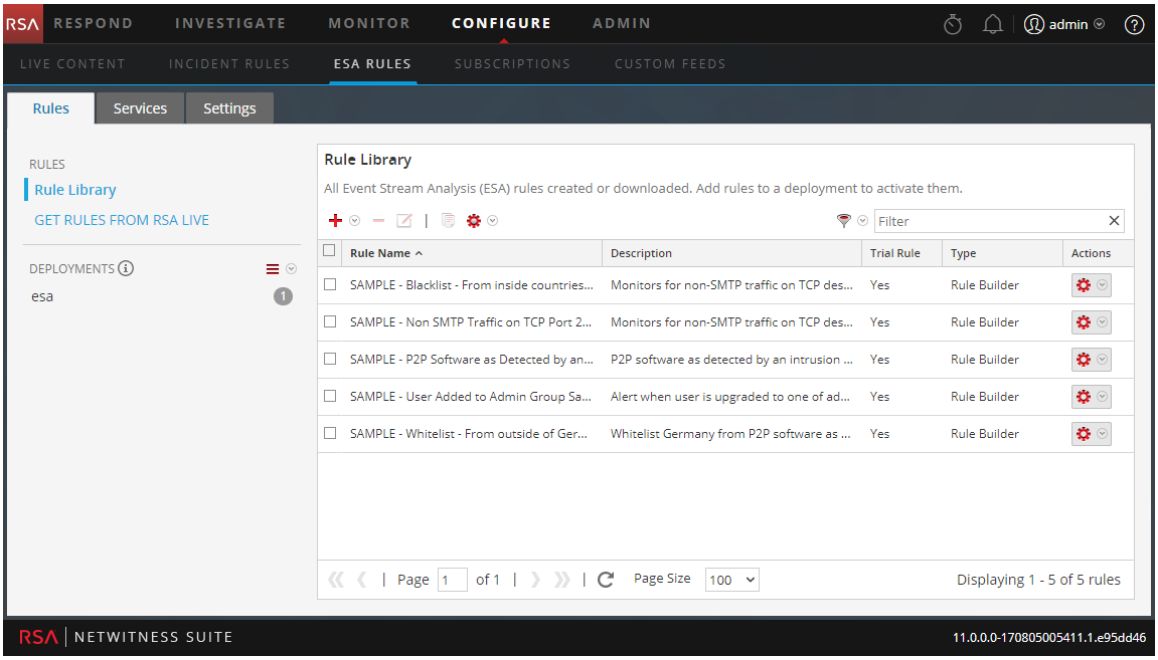
If you exported a rule in an earlier version of NetWitness Suite, the following conditions apply when you import the rule in version 10.5 or later:

- Exported in version 10.3 – You cannot import rules to version 10.5 or later.
- Exported in version 10.4 – Rule behavior depends if cross-correlation is disabled, which is the default, or enabled:
  - Disabled – You can import rules to version 10.5 or later.
  - Enabled – You must restart NetWitness Suite or make a minor change to the rule, save, remove the minor change and save again. Either procedure generates the forwarding rule that the 10.5 or later cross-site correlation feature requires.

Procedures

Import ESA Rules

1. Go to **CONFIGURE >ESA Rules > Rules** tab.
   The Rules tab is displayed.



2. In the **Rules Library** toolbar, select > **Import**.
   The Import ESA Rules dialog is displayed.

3. Click **Browse** to browse and select the file containing the ESA rules.
4. Click **Import**.

Export

1. Select an ESA rule or multiple rules and select > **Export** in the Rule Library toolbar.
   A warning dialog is displayed.
2. Click **Yes**.
   The Export Rules dialog is displayed.

3. In the **Enter File Name** field, type a filename for the file with the ESA rules and click **Export**.
   The file is exported as a binary file to your machine.

Note: The binary file cannot be edited.

Choose How to be Notified of Alerts

This topic explains the different notification methods and how to add a notification method to a rule. Administrator, SOC Manager or DPO role permissions are required for all tasks in this section.

When a rule triggers an alert, ESA can send a notification in the following ways:

- Email
- SNMP
- Syslog
- Script

To configure a notification, you configure these components:

- Notification server – After you configure a notification server, you can add it to a rule. When the rule triggers an alert, the rule will use that server to send alert notifications.
- Notifications – These are the outputs, which can be email, script, SNMP, and Syslog. When you design a rule, you can specify the notification for an alert.
- Templates – The format of an alert notification is defined in a template.

Alert suppression and alert rate regulation are two features that Event Stream Analysis provides. Alert suppression ensures that multiple emails are not sent out for the same alert. For example, consider a rule to detect failed user logins. If you set the alert suppression to three minutes, you will see only the alerts generated in that time frame. This is fewer than the number of alerts you would see without alert suppression. Some alerts can be duplicates. With alert suppression, emails are not sent for duplicate alerts. This ensures the inbox is not flooded with redundant alert notifications.

Alert rate regulation is a preventive measure to ensure that alerts from misconstrued rules do not flood the system. This ensures that ESA does not send more than the configured limit of emails within one minute.

Notification servers, notifications, and templates are configured in the Administration System view. For more information, see "Configure Notification Servers", "Configure Notification Outputs", and "Configure Templates for Notifications" in the *System Configuration Guide*.

Notification Methods

When a rule triggers an alert, ESA can send a notification in the following ways:

- Email
- SNMP
- Syslog
- Script

Email Notifications

Event Stream Analysis can send notifications to users through email about various system events.

To configure these email notifications, you need to:

- Configure the SMTP email server as an output provider. For instructions, see "Configure the Email Settings as Notification Server" in the *System Configuration Guide*.
- Set up an email account to receive notifications. For instructions, see "Configure Email as a Notification" in the *System Configuration Guide*.
- Configure a template for email notification. For instructions, see "Configure a Template" in the *System Configuration Guide*.

SNMP

Event Stream Analysis can send events as an SNMP trap to a configured SNMP trap host.

Note:
The MIB file **NETWITNESS-MIB.txt** is located on the ESA RPM at the following location */usr/share/snmp/mibs*. With the MIB file, you will be able to identify the SNMP alerts trigerred from ESA. And, the Trap OID value for ESA is 20.

To configure these SNMP notifications, you need to:

- Configure SNMP trap host settings as an output provider. For instructions, see "Configure the SNMP Settings as Notification Server" in the *System Configuration Guide*.
- Configure SNMP trap settings as an output action. For instructions, see "Configure SNMP as a Notification" in the *System Configuration Guide*.
- Configure a template for SNMP. For instructions, see "Configure a Template" in the *System Configuration Guide*.

Syslog

Event Stream Analysis can send events and consolidate logs in Syslog format to a Syslog server.

To configure these Syslog notifications, you need to:

- Configure Syslog server settings as an output provider. For instructions, see "Configure the Syslog Settings as Notification Server" in the *System Configuration Guide*.
- Configure Syslog message format as an output action. For instructions, see "Configure Syslog as a Notification" in the *System Configuration Guide.*
- Configure a template for Syslog. For instructions, see "Configure a Template" in the *System Configuration Guide*.

Script Alerter

Apart from the alert notifications ESA allows users to run scripts in response to ESA alerts.

Scripts enable you to do custom integration with applications that exist in your environment. For example, if you want to open an incident ticket from an application when a specific alert is triggered, Script Alerter lets you write a script that calls the application API and has ESA invoke it when the specific ESA rule is triggered. You can configure a FreeMarker template to define what details you want to extract from the output of the ESA rule and pass it as command line arguments to the script.

To use the Script Alert, you need to:

- Configure the user identity and other details that are required to execute the script. For instructions, see "Configure Script as a Notification Server" in the *System Configuration Guide*.
- Define the Script. For instructions, see "Configure Script as a Notification" in the *System Configuration Guide*.
- Configure a template for the script. For instructions, see "Configure a Template" in the *System Configuration Guide*.

Add Notification Method to a Rule

This topic tells administrators how to add a notification, such as email, to a rule. ESA uses the notification method when it generates an alert for an event that meets rule criteria.

You add a notification to a rule so ESA can let you know when a rule triggers an alert. Although the notification fields are not required, it is a best practice to add a notification to a rule.

When you add a notification method to a rule, you select the following information:

- Output
- Notification
- Notification Server
- Template

Prerequisites

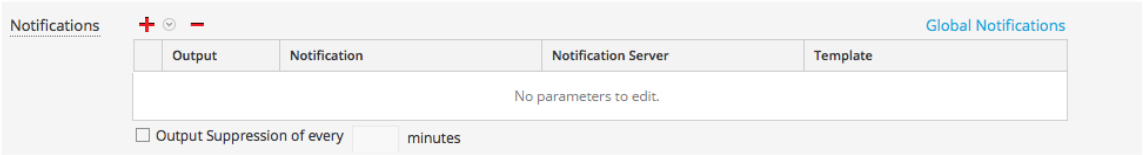
- Your role must have permission to manage rules.
- The rule must exist.
- The notification method must be configured with a supported server and template:
  Go to **ADMIN** > **System > Global Notifications**.

  For detailed procedures, see the *System Configuration Guide*.

Procedure

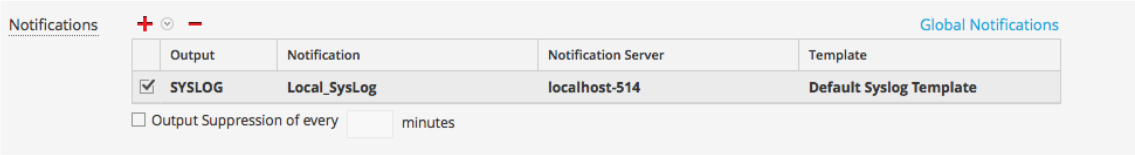
To add a notification method to a rule:

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
2. In the **Rule Library**, click  to add a new rule or select an existing rule and click  .
   Depending on the rule type, the Rule Builder or Advanced EPL tab is displayed.
   The Notifications section is the same for both tabs.



3. Click  and select the **Output** for the alert:
   - Email
   - SNMP
   - Syslog
   - Script
4. Double-click the **Notification** field and select the name of a previously configured output.
   For example, Level 1 Analyst could be the name of an email notification that goes to the L1-Analysts email distribution group.
5. Double-click the **Notification Server** field and select the server that sends the notification.

6. Double-click the **Template** field and select a format for the alert.
   The following figure shows the settings for a Syslog notification.

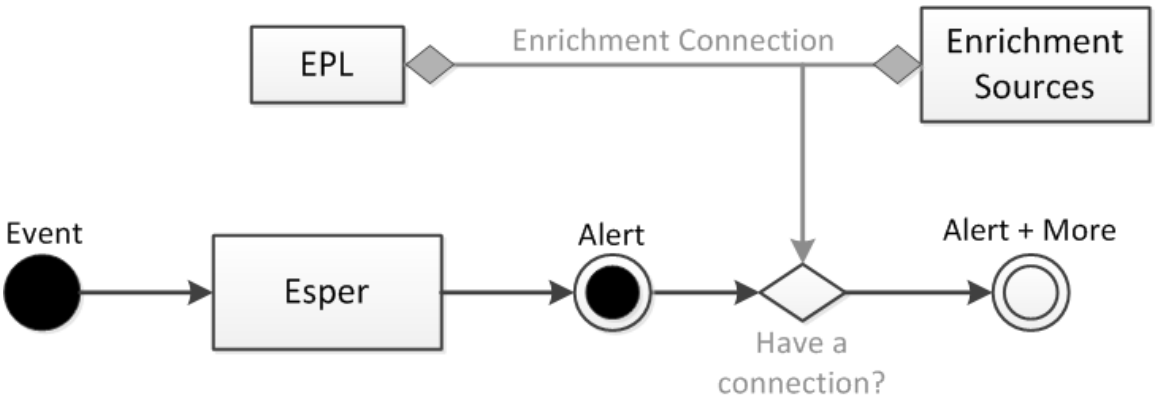


7. If you want to specify frequency, select **Output Suppression**, then enter the number of **minutes**.
8. If you want to add another notification, repeat steps 3-7.
9. Click **Save**.
   When ESA generates an alert for an event that matches the rule criteria, you will be notified of the alert via each notification method added to the rule.

Add a Data Enrichment Source

This topic tells how to add a previously configured enrichment source to a rule. When ESA creates an alert, information from the source gets included in it.

Enrichments provide the ability to include contextual information into correlation logic and alert output. Without enrichments, all information included in an ESA alert is from a Core service. With enrichments, you can request for look ups into a variety of sources and include the results into the outgoing alerts. The following figure illustrates the enrichment feature.



Enrichment configuration is made up of two logical units:

- Enrichment Sources – These are data stores of contextual information.
- Enrichment Connections – These act as connectors between alert meta and source columns.

ESA allows you to make connections between Event Processing Language (EPL) statements and enrichment sources. Once the connections are established, the system joins the selected fields from the alert output with the information in the sources and uses the matching data to enrich the alert that is sent out. ESA can connect with the following sources:

- Esper Named Windows
- Relational Database tables
- MaxMindGeoIP Database
- RSA Warehouse Analytics Watchlists

Note: The geoIP enrichment source can neither be created nor deleted. It is provided out of the box to the user.

Sample Rule with Enrichment

The following sample rule illustrates the enrichment feature provided by ESA:

```
@RSAAlert @Name("simple") SELECT * FROM CoreEvent(ec_theme='Login
Failure')
```

The rule generates an alert for every logon failure and thus if the following (simplified) event stream is received at ESA:

sessionid	ec_ theme	username	ip_src	ip_dst	host_dst
1	Login Success	dshrute	23.xx.23x.16		
2	Login Failure	jhalpert	23.xx.23x.16	31.1x.x9.1x8	www.facebook.com

An alert with the following constituent `events` might be generated in response to the second session:

```
{
     "events": [
         {
             "username": "jhalpert",
             "host_dst": "www.facebook.com",
             "ip_dst": "31.1x.x9.1x8",
             "sessionid": 2,
             "ec_theme": "Login Failure",
             "esa_time": 1406148964130,
             "ip_src": "23.xx.23x.16"
         }
     ]
 }
```

The JSON output shows all the information available for inclusion into an ESA notification using an appropriate FreeMarker
template. For instance, the template expression `${events[0].username}` would evalute to `jhalpert`.

With enrichments, the same module, with the same event stream, can generate the alert shown below. The system
can make multiple enrichment connections and pull contextual data to make the alert more meaningful.

For example:
`${events[0]["RSADataScienceLookup"][0].score}` gives the **"risk"** score of the destination domain computed by the RSA Warehouse Analytics module while `${events[0]["orgchart"][0].supervisor}` gives the name of the supervisor of the employee that the alert pertains to (pulled from an HR database) and `${events[0]["LoginRegister"][0].username}` gives the name of the user with the last successful logon from the same `ip_src` (using a stream based Named Window).


```
{"events": [
     {
         "username": "jhalpert",
         "host_dst": "www.facebook.com",
         "GeoIpLookup": [
         {
             "city": "Cambridge",
             "longitude": -71,
             "countryCode": "US",
             "areaCode": 617,
             "metroCode": 506,
             "region": "MA",
             "dmaCode": 506,
             "ipv4Obj": "/23.62.236.16",
             "countryName": "United States",
             "postalCode": "02142",
             "ipv4": "23.62.236.16",
             "latitude": 42,
             "organization": "Verizon Business"
         }
     ],
     "RSADataScienceLookup": [
         {
             "model_id": "suspiciousDomains_1",
             "_id": "EXEC_BATCH_1_20140630153740_facebook.com",
             "score": 10,
             "key": "www.facebook.com"
         }
     ],
     "orgchart": [
         {
             "supervisor": "mscott",
             "name": "James Halpert",
             "extension": 3692,
             "location": "Scranton",
             "department": "Sales",
```

```
            "id": "jhalpert"
        }
    ],
    "ip_dst": "31.13.69.128",
    "sessionid": 2,
    "LoginRegister": [
        {
            "username": "dshrute",
            "ip_src": "23.62.236.16"
        }
    ],
    "ec_theme": "Login Failure",
    "esa_time": 1406155218912,
    "ip_src": "23.62.236.16"
    }
]}
```


Configure a Database Connection

This topic provides information to configure a connection to an external database that can provide additional information in alerts. You configure a database connection so you can then configure the database as an enrichment source, to add further details to alerts. There are three steps in the process:

1. Configure a connection to a database.
2. Configure the external database as an enrichment source.
3. Add the enrichment source to a rule

This topic explains Step 1.

Example

This example illustrates how adding a database as an enrichment source adds value to alerts.

A rule detects users that attempt to sign up for a stealth email service. Twenty-five users match the rule criteria. Without the enrichment, the alert contains 25 User IDs. With the enrichment, the alert also includes the following information for each User ID:

- Name
- Title
- Department
- Office Location

Dependencies

When you configure a database, the following conditions apply:

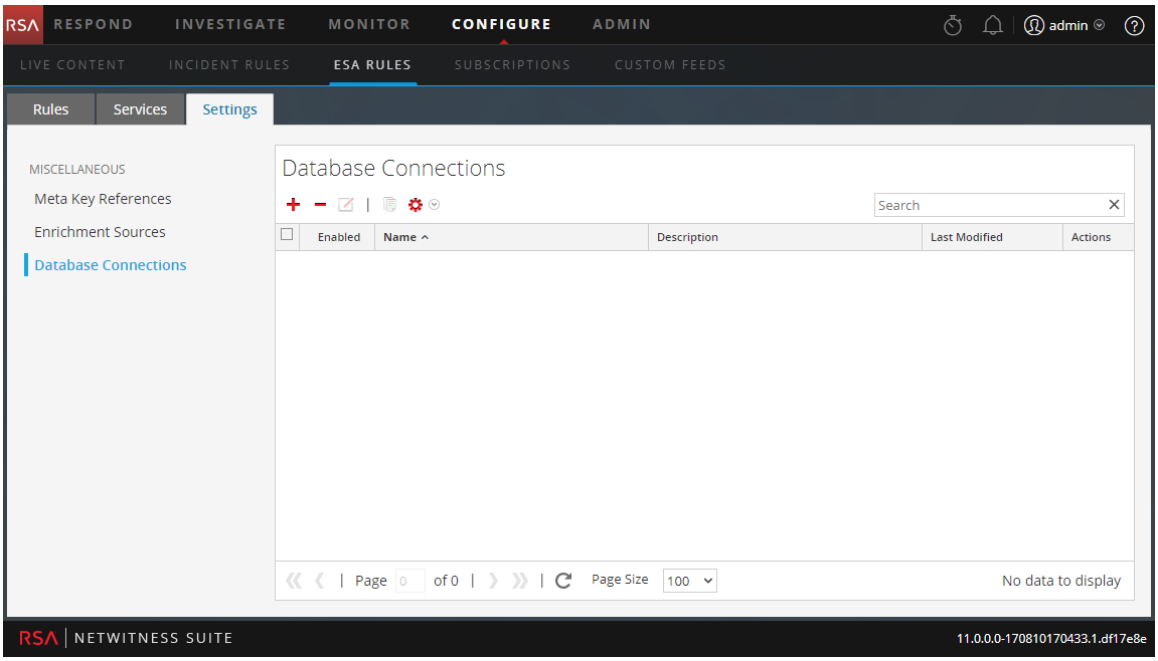
- A reference to the database is deployed on every ESA, even if the ESA does not deploy rules that use the database as an enrichment source.
- If the server that hosts the database goes down, it impacts a deployment.
  - An active deployment will continue to gather data and run rules but enrichments will not appear in alerts.
  - A new deployment will fail until you restart the host.

Procedure

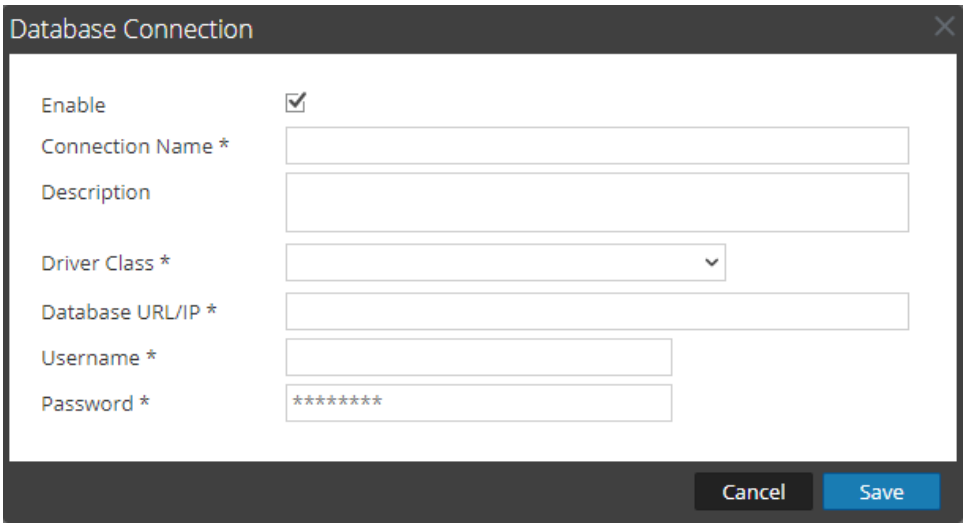
To configure a database connection:

1. Go to **CONFIGURE > ESA Rules**.
2. Click the **Settings** tab.
3. In the options panel, select **Database Connections**.

   The Database Connections panel is displayed.



4. Click + to add a database connection.



5. In the **Database Connection** dialog, provide the following information.

Field	Description
Enable	Select Enable to enrich the alert with additional data. By default, Enable is selected. Deselect Enable to exclude additional data from the alert.
Connection Name	Type a name to identify the connection. When you add a database as an enrichment source, this name appears in the list of Database Connections.
Description	(Optional) Type a brief description about the database connection.
Driver Class	Select an appropriate driver class for the database. Two drivers come with NetWitness Suite, MongoDB and Postgres.
Database URL or IP address	Type the URL or the IP address of the database to configure.
Username	Type the username to access the Database.
Password	Type the password to access the Database.

6. Click **Save**.

For related information, see Settings Tab.

Enrichment Sources

This topic explains options for adding an external data source to provide additional information in alerts. Enrichment sources provide additional information in alerts. For example, a database can provide a name, department, and office location if a user matches rule criteria. There are three types of enrichment sources:

- External DB Reference
- In-Memory Table
- Warehouse Analytics

Configure a Database as Enrichment Source

You can configure a database as an enrichment source so you can add it to a rule. Then the Esper engine that analyzes events can access the information in the database to provide additional information in the alert.

For example, a rule detects users that attempt to sign up for a stealth email service. Twenty-five users match the rule criteria. The alert contains 25 User IDs. An external database would enhance the alert by providing the following additional information for each User ID:

- Name
- Title
- Department
- Office Location
- Reports To

You can edit, duplicate, import or export a database connection.

Prerequisites

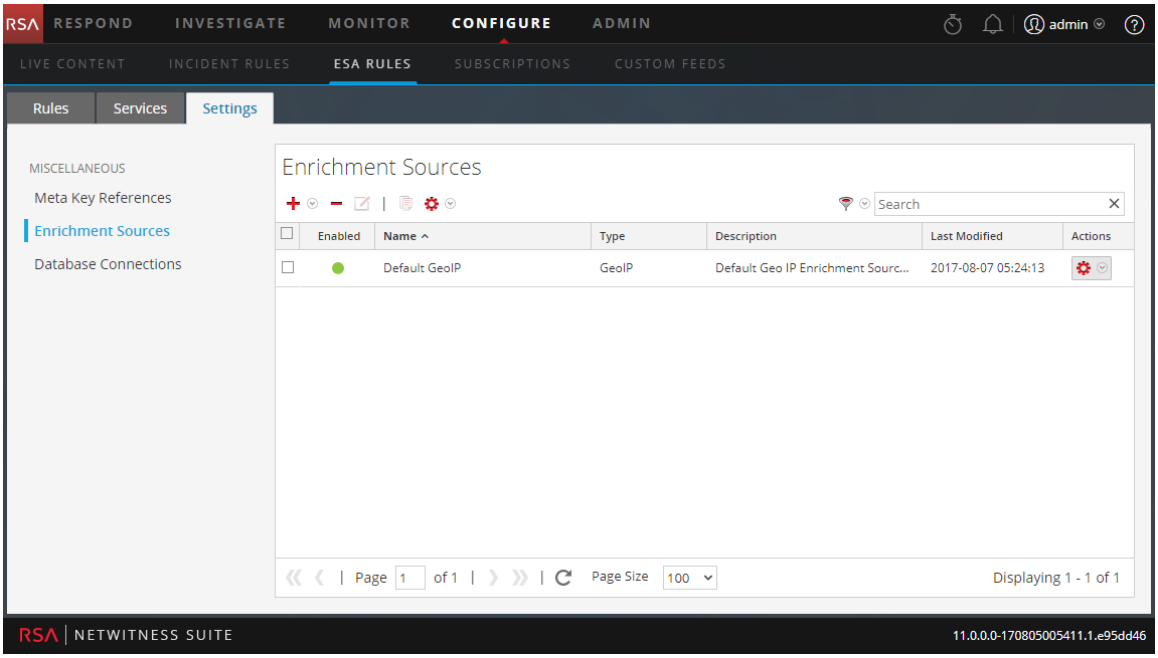
You must configure a database connection. For more information, see Configure a Database Connection.

Procedure

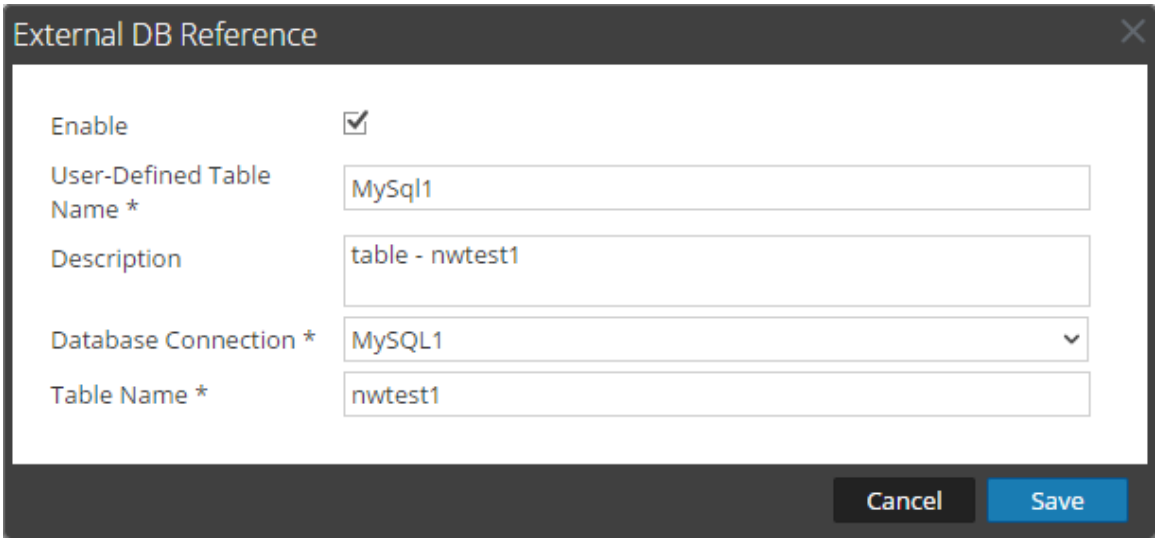
To configure database as an enrichment source:

1. Go to **CONFIGURE > ESA Rules**.
2. Click the **Settings** tab.
   The Settings tab is displayed.

3. In the options panel, select **Enrichment Sources**.
   The Enrichment Sources panel is displayed.



4. From the  drop-down menu, select **External DB Reference**. You have to add a DB reference in order for the DB to be listed.
   The External DB Reference dialog is displayed.



5. Select **Enable** to enrich alert with additional data. This is selected by default. If disabled, the alert will not be enriched with additional data.
6. In the **User-Defined Table Name** field, type a name to identify or label the database configuration.

7. In the **Description** field, type a brief description about the database configuration.
8. In the **Database Connection** drop-down menu, select the database connections defined.
9. In the **Table Name** field, enter database table name.
10. Click **Save**.

For details on parameters and their descriptions, see Settings Tab.

Configure In-Memory Table as Enrichment Source

This topic provides instructions on how to configure an in-memory table. When you configure an in-memory table, you upload a .CSV file as an input to the table. You can associate this table with a rule as an enrichment source. When the associated rule generates an alert, ESA will enrich the alert with relevant information from the in-memory table.

For example, a rule could be configured to detect when a user tries to download freeware and to identify the person by user ID in the alert. The alert could be enriched with additional information from an in-memory table that contains details such as full name, title, office location and employee number.

An in-memory table is ideal for handling lightweight data. It is easy to set up and requires less maintenance than a database. For example, the AllTech Company is a small organization so the system administrator can maintain employee information in a .CSV file. If AllTech grows into a very large company, the administrator would have to configure an external database reference as an enrichment and associate the database with a rule.

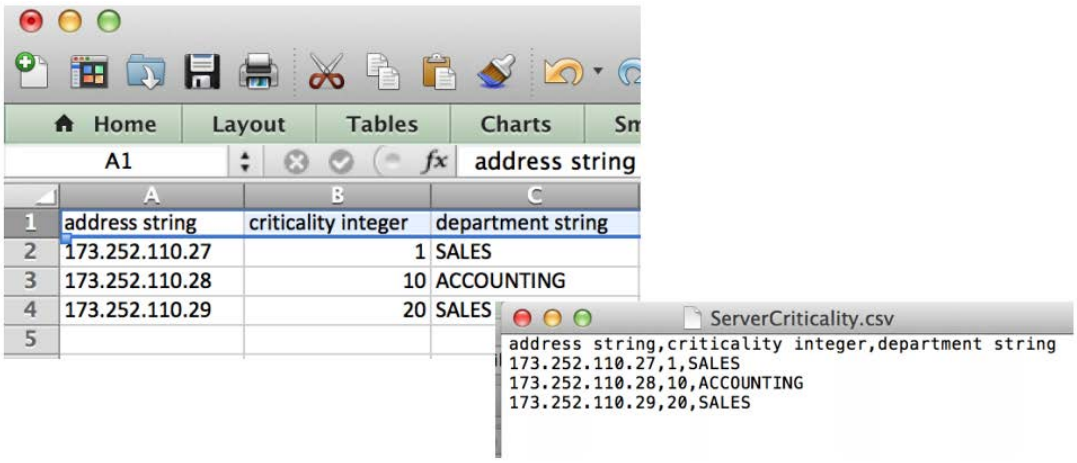
Prerequisites

The column name in the .CSV file cannot have whitespace characters.

For example *Last_Name* is correct, and *Last Name* is incorrect.

The .CSV file must begin with a header line that defines fields and types.
For example, *address string* would define the header field as *address*, and the type as *string*.

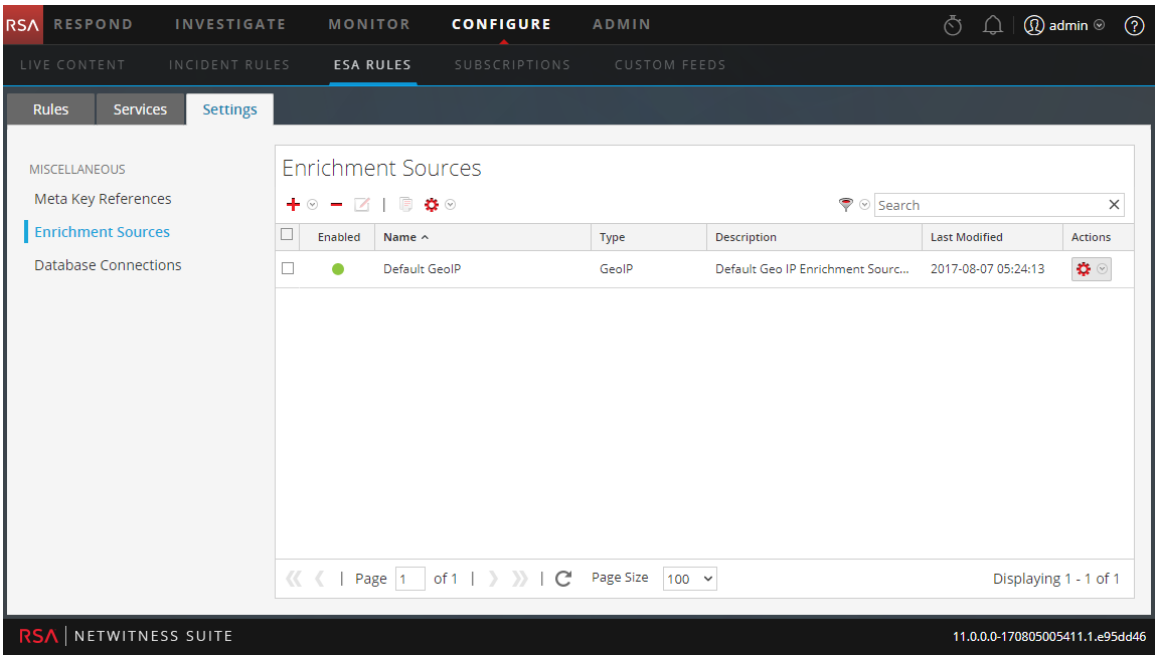
The following shows a valid .CSV file represented as a .CSV and as a table.



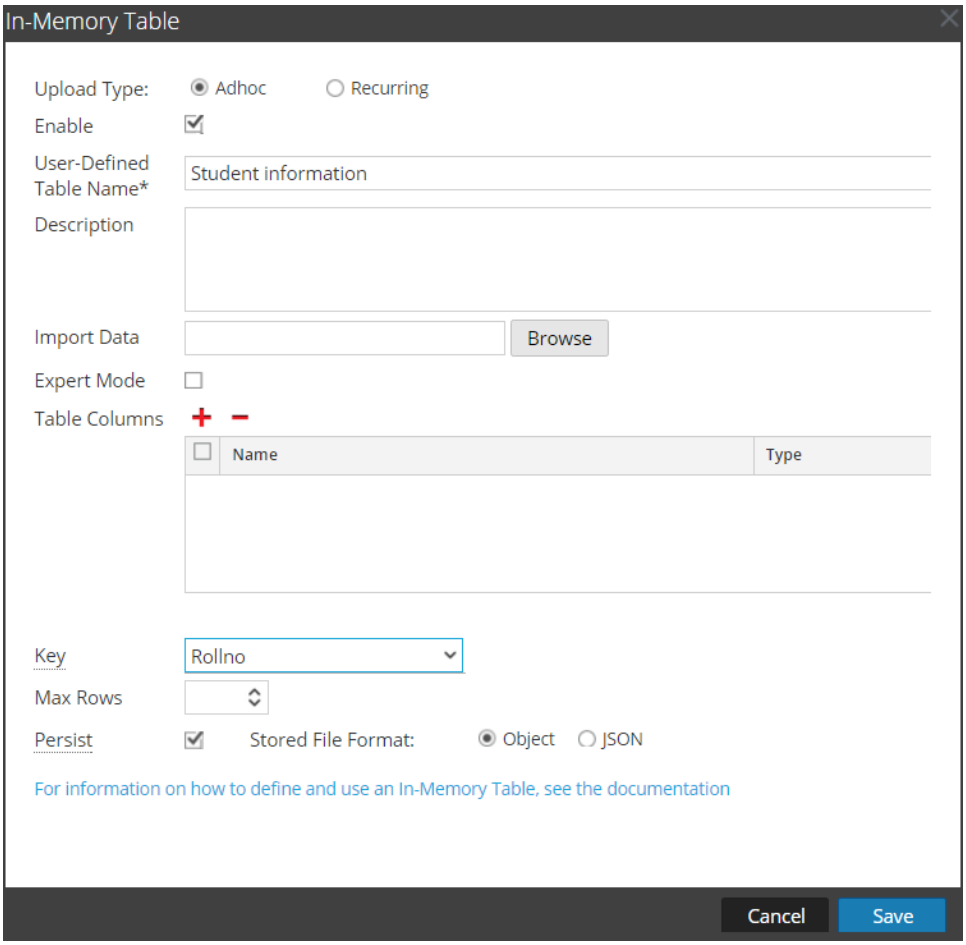
Procedures

Configure an Ad hoc In-Memory Table

1. Go to **CONFIGURE > ESA Rules**.
   The Configure view is displayed with the ESA Rules tab open.
2. Click the **Settings** tab.
3. In the options panel, select **Enrichment Sources**.



4. In the **Enrichment Sources** section, click + > **In-Memory Table**.



5. Describe the in-memory table:

   a. Select **Ad hoc**.

   b. By default, **Enable** is selected. When you add the in-memory table to a rule, alerts will be enriched with data from it.
   If you add an in-memory table to a rule but do not want alerts to be enriched, deselect the checkbox.

   c. In the **User-Defined Table Name** field, type a name, such as Student Information, for the in-memory table configuration.

   d. If you want to explain what the enrichment adds to an alert, type a **Description** such as:
   When an alert is grouped by Rollno, this enrichment adds student information, such as name and marks.

6. In the **Import Data** field, select the .CSV file that will feed data to the in-memory table.

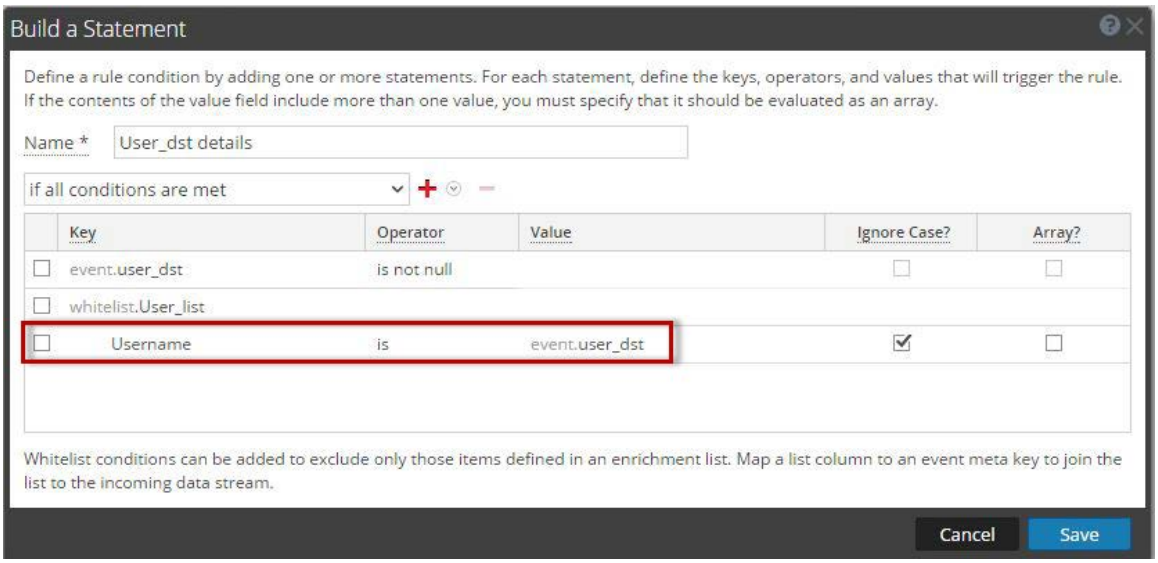
7. If you want to write an EPL query to define an advanced in-memory table configuration, select **Expert Mode**.
   The Table Columns are replaced by a **Query** field.
8. In the **Table Columns** section, click + to add columns to the in-memory table.
9. If a valid file is selected in the Import Data field, the columns populate automatically.

Note: If you selected Expert mode, a Query field is displayed instead of Table Columns.

10. In the **Key** drop-down menu, select the field to use as the default key to join incoming events with the in-memory table when using a CSV-based in-memory table as an enrichment. By default, the first column is selected. You can also later modify the key when you open the in-memory table in enrichment sources.
11. In **Max Rows** drop-down menu, select the number of maximum number of rows that can reside in the in-memory table at a particular instance.
12. Select **Persist** to preserve the in-memory table on disk when the ESA service stops and to re-populate the table when the service restarts.
13. In **Stored File Format** field, do one of the following:
    - Select **Object**, if you want to store the file in a binary format.
    - Select **JSON**, if you want to store the file in a text format.
      By default, **Object** is selected.
14. Click **Save**.
    The adhoc in-memory table is configured. You can add it to rule as an enrichment or part of the rule condition. See Add an Enrichment to a Rule.

When you add an in-memory table, you can add it to a rule as an enrichment or as a part of the rule condition. For example, the following rule uses an in-memory table as a part of the rule condition to create a whitelist, and it also uses an in-memory table of details in the user_dst file to enrich the alert that is displayed.

The rule shows the in-memory table as a whitelist rule condition:



Next, the alert is enriched with the User_list in-memory table:

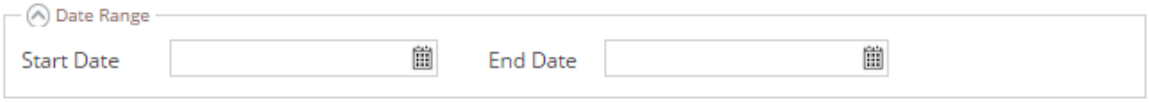
Therefore, the user_dst in-memory table is used to create a whitelist, and it is also used to enrich the data in the alert if the alert is triggered.

Add a Recurring in-Memory Table

1. Go to **CONFIGURE > ESA Rules**.
   The Configure view is displayed with the ESA Rules tab open.
2. Click the **Settings** tab.
3. In the options panel, select **Enrichment Sources**.
4. Click > **In-Memory Table**.
5. Describe the in-memory table:
   a. Click **Recurring**.
   b. By default, **Enable** is selected. When you add the in-memory table to a rule, alerts will be enriched with data from it.
   If you add an in-memory table to a rule but do not want alerts to be enriched, deselect the checkbox.
   c. In the **User-Defined Table Name** field, type a name, such as Student Information, for the in-memory table configuration.

d. If you want to explain what the enrichment adds to an alert, type a **Description** such as: When an alert is grouped by Rollno, this enrichment adds student information, such as name and marks.

6. Type the URL of the .CSV file that will feed data to the in-memory table. Click Verify to validate the link and populate the columns in the .CSV file. You can add or remove columns using the plus or minus button.
7. If the server is configured behind another server, select **Use Proxy**.
8. If the server requires logon credentials, select **Authenticated**
9. For **Recur Every**, indicate how frequently ESA must check for the most recent .CSV:
   a. Select Minute(s), Hour(s), Day(s), or Week.
   b. If you select Week, select a day of the week.
   c. Click **Date Range** to select a **Start Date** and **End Date** for the recurring schedule.



10. In the **Key** drop-down menu, select the field to use as the default key to join incoming events with the in-memory table when using a CSV-based in-memory table as an enrichment. By default, the first column is selected. You can also later modify the key when you open the in-memory table in enrichment sources.
11. In **Max Rows** drop-down menu, select the number of rows that can reside in the in-memory table at a particular instance.
12. Select **Persist** to preserve the in-memory table on disk when the ESA service stops and to re-populate the table when the service restarts.
13. In **Stored File Format** field, do one of the following:
    - Select **Object**, if you want to store the file in a binary format.
    - Select **JSON**, if you want to store the file in a text format.
      By default, **Object** is selected.
14. Click **Save**.
    The recurring in-memory table is configured. You can add it to rule as an enrichment or part of the rule condition. See Add an Enrichment to a Rule.

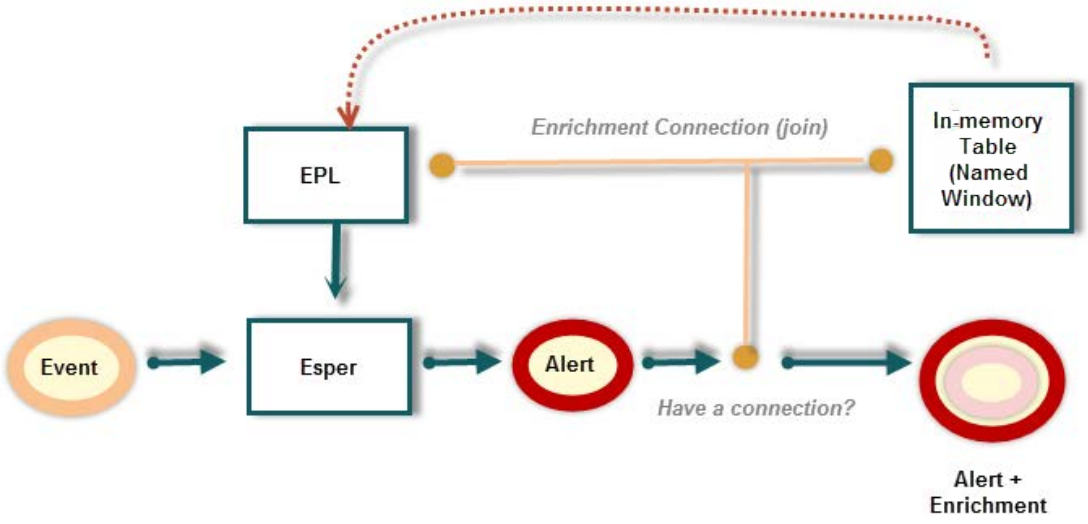
Configuring an Esper Query as an Enrichment Source

When you use "expert mode," you can create an enrichment source or named window based on an Esper query. This allows you to have more control over the content and create more dynamic content. When you do this, an EPL query constructs the named window to capture interesting state from event stream.

Workflow

The following shows the workflow for creating a query using a named window:

1. The event is sent to the Esper Engine.
2. An EPL query is generated.
3. An alert is triggered.
4. The query checks to see if there is a connection between the event and the Named Window.
5. If there is a connection, the query that populates the Named Window is run and populated.
6. The content from the Named Window is added to the alert content and sent or displayed (depending on your settings).



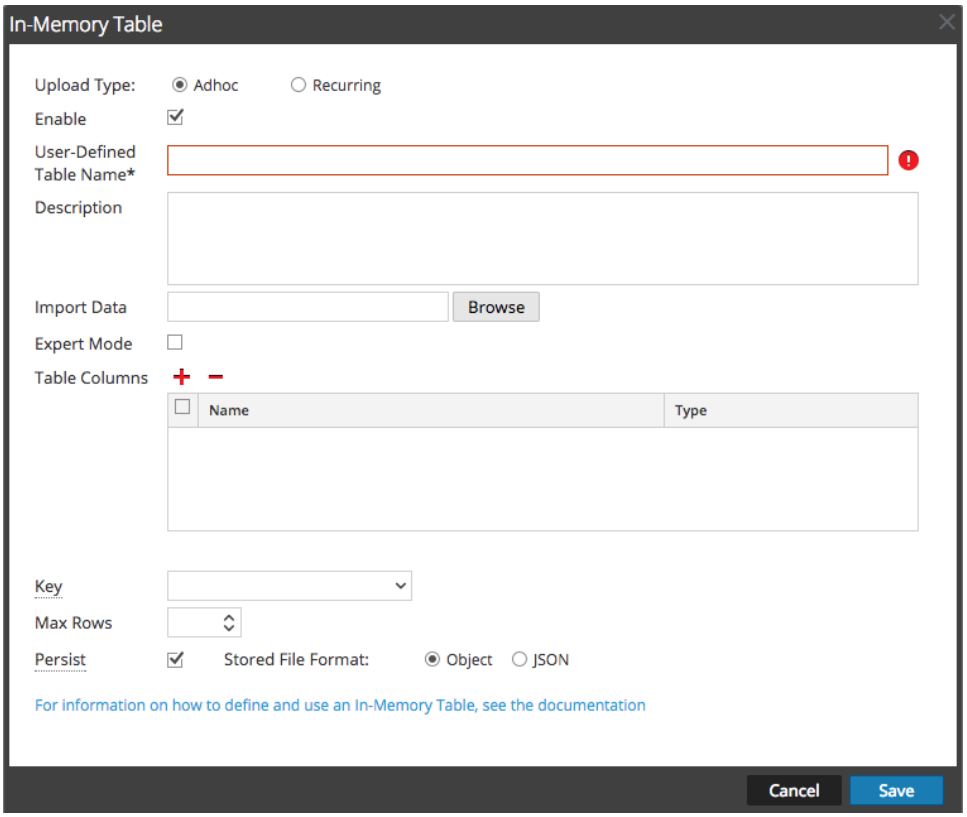
Prerequisites

- The meta used in the EPL statement must exist in the data.
- You must create well-formed EPL statements.

Procedure

Configure an In-Memory Table Using an EPL Query

1. Go to **CONFIGURE > ESA Rules**.
   The Configure view is displayed with the Rules tab open.
2. Click the **Settings** tab.
3. In the options panel, select **Enrichment Sources**.
4. In the **Enrichment Sources** section, click > **In-Memory Table**.



5. Select **Adhoc**.
   By default, Enable is selected. When you add the in-memory table to a rule, alerts will be enriched with data from it.
6. In the **User-Defined Table Name** field, type a descriptive name to describe the in-memory table.
7. If you want to explain what the enrichment adds to an alert, enter information in the **Description** field.
   This description displays when you view the list of enrichments from the Enrichment Sources

view, so it's a good idea to enter a thorough description as a best practice. Doing this allows other users to understand the content of the enrichment without opening it to examine its contents.

8. Select **Expert Mode** to define an advanced in-memory table configuration by writing an EPL query.
   The Table Columns are replaced by a **Query** field.
9. Select **Persist** to preserve the in-memory table on disk when the ESA service stops and to re-populate the table when the service restarts.
10. Enter the EPL query in the **Query** field. The query should be well-formed, and it's a good idea to test it before entering it in the field.
11. Click **Save**.

Example

For example, you created a rule that searches for five failed attempted logins followed by a successful login. When that rule is triggered, you may want the notification to contain information about the last user logged into the system when this successful login occurred. To add this enrichment to the notification, you might choose to create a stream-based in-memory lookup table that is populated from incoming events to maintain a mapping of IP addresses to the last user logged in from that address. To do this, you create an enrichment using a query as your source.

Step 1: Create Your Rule

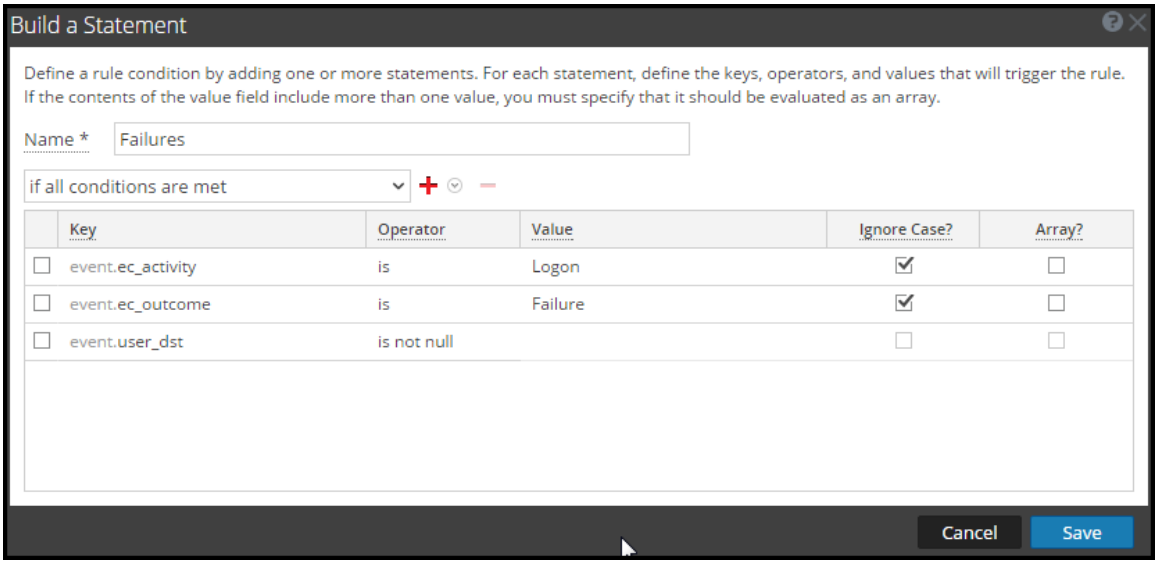
First, you need to create your correlation rule. In this case, you create failure and success rule conditions, and group by the ip_src.

Rule Condition	Description
Failures	This condition searches for five failed logins with a "followed by" connector, meaning that the condition (Failures) must be followed by the next condition (Success).
Success	This condition searches for one successful login.

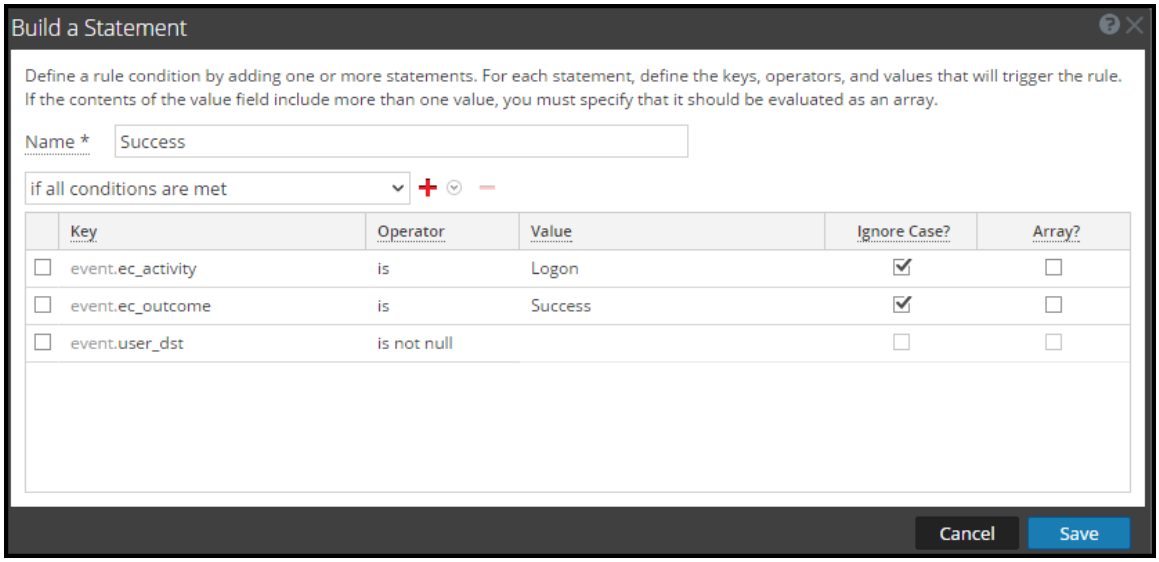
Rule Condition	Description
GroupBy: ip_src, device class	The GroupBy field ensures that all the previous conditions are grouped by the ip_srcand device class. This is important to the construction of the rule because the rule attempts to find a case where a user has attempted to log into the same destination account multiple times, and finally logged in successfully. Grouping by device class ensures that the user logged in from the same machine attempted to log into an account multiple times. The rule may give unexpected results if you do not group the results.
Occurs within 5 minutes	The time window for the events to occur is five minutes. If the events occur outside of this time window, the rule does not trigger.
Event Sequence: Strict	The event sequence is configured for a strict pattern match. This means that the pattern must match exactly as it is specifiedwith no intervening events.

For the rule conditions, you create the following statements:

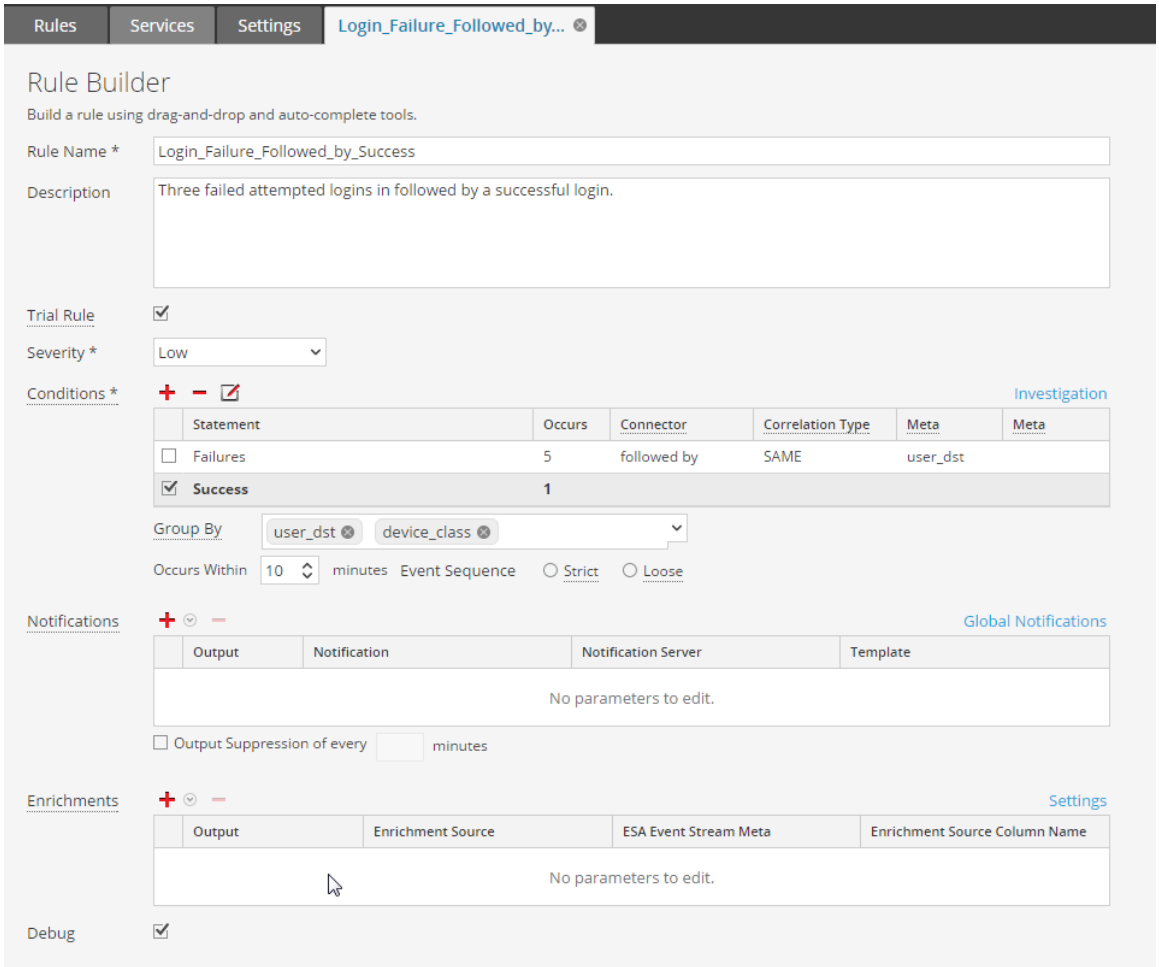
- The "Failures" statement searches for failed login attempts:



- The "Success" statement searches for one successful login:



- Combined, you have the following correlation rule:



Step 2: Create the Enrichment

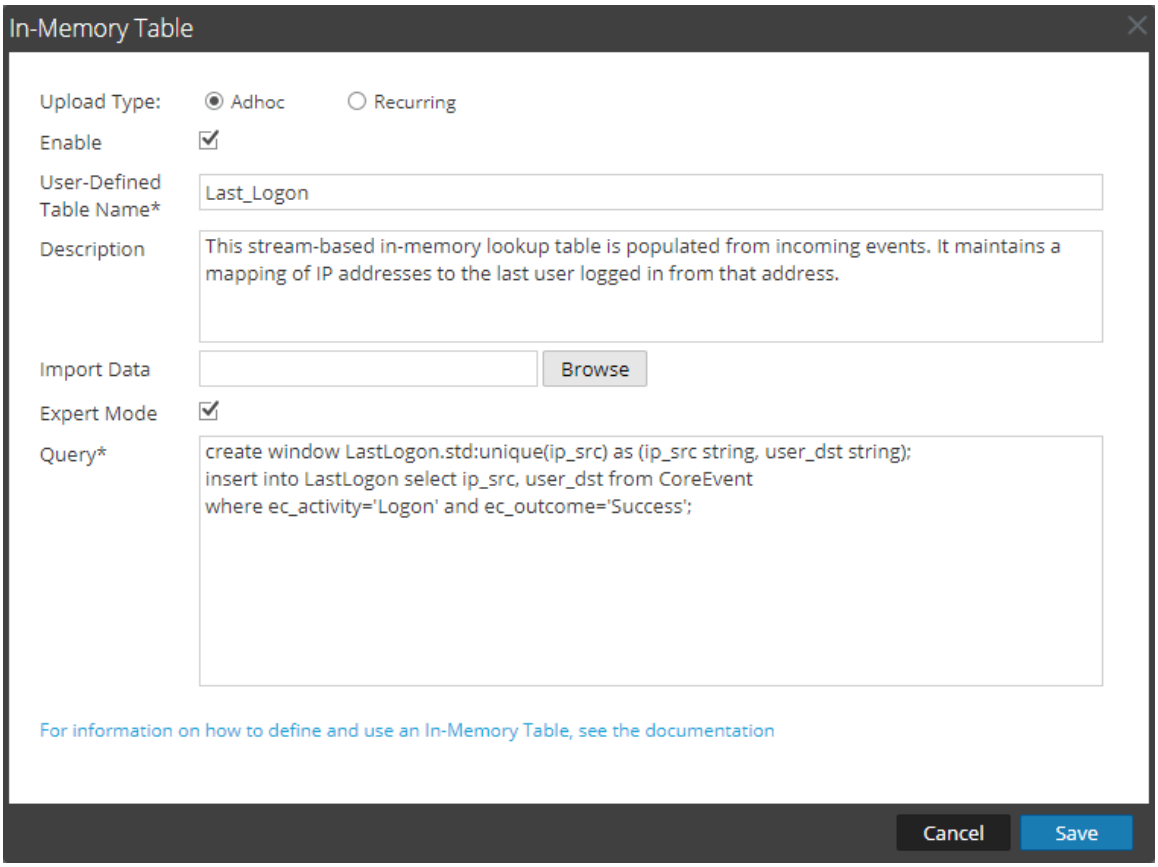
Now that you have created your rule, you need to create the enrichment to add to the notification output. Follow the steps above to create the enrichment, name it *Last_Logon*, and add the following query:

```
create window LastLogon.std:unique(ip_src) as (ip_src string, user_dst string);

insert into LastLogon select ip_src, user_dst from CoreEvent

where ec_activity='Logon' and ec_outcome='Success';
```

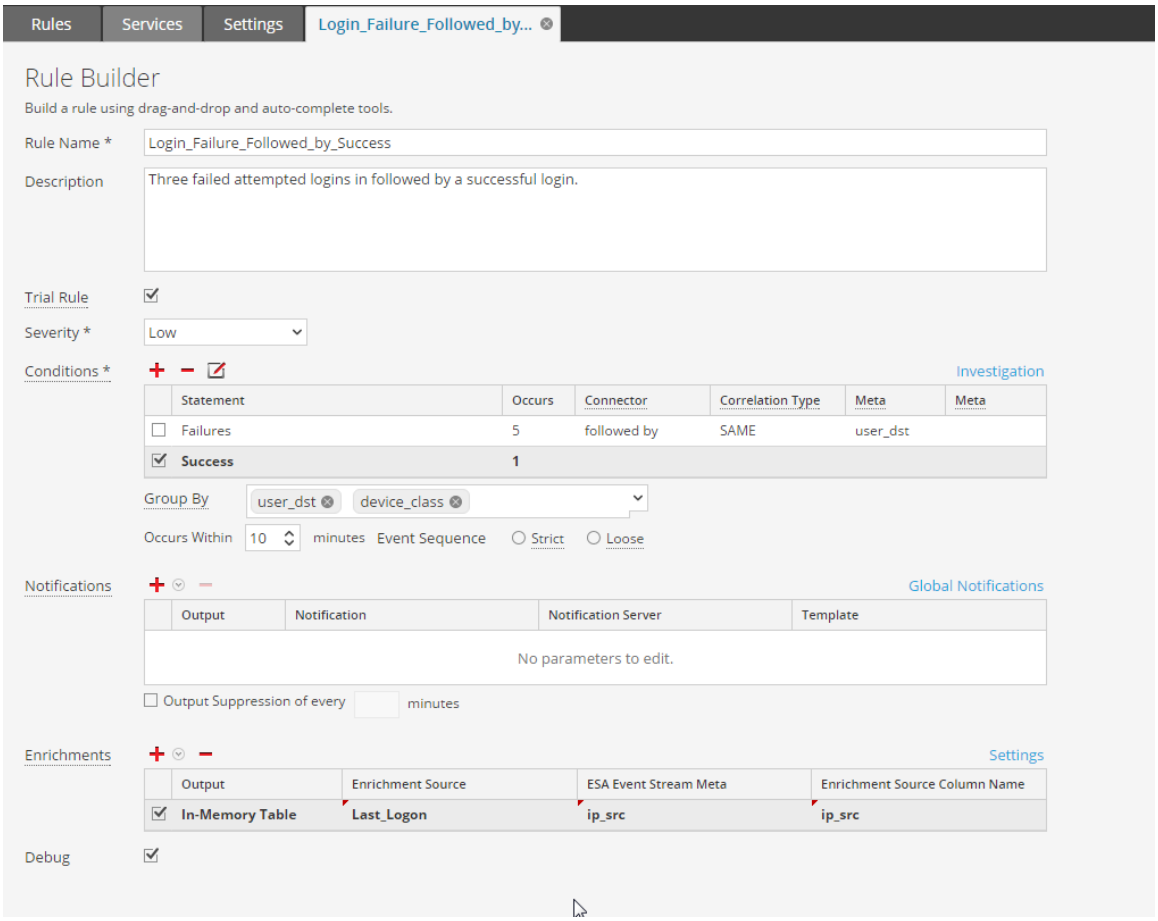
The enrichment should look like the following:



Step 3: Add the Enrichment to the Rule

Now that you have created your basic rule and your enrichment, you'll need to add the enrichment to the rule and join (or connect) the enrichment to the meta in the rule.

Open the Login_Failure_Followed_by_Success rule for editing.



Field	Enter	Description
Output	In-Memory Table	The In Memory Table option creates a Named Window, which can be populated with the EPL query data.
Enrichment Source	Last_Logon (the enrichment you created above).	This is the stream-based in-memory lookup table that is populated from incoming events to maintain a mapping of IP addresses to the last user logged in from that address.
ESA Event Stream Meta	ip_src	This is an event stream meta that you can join to the enrichment data you are populating. Essentially, ip_src is the join condition .
Enrichment Source Column Name	ip_src	This is the meta from the enrichment that you can join to the event stream data. It must be the same as join condition from the Event Stream Meta field.

Once you have added the enrichment, you can save the rule.

When the rule is triggered, the ESA runs the query in the enrichment and populates the Named Window with the data. If the data in the Named Window matches the join condition, the data is added to the output you can view in Email, SNMP, Syslog or Script, depending on how you configured notifications.

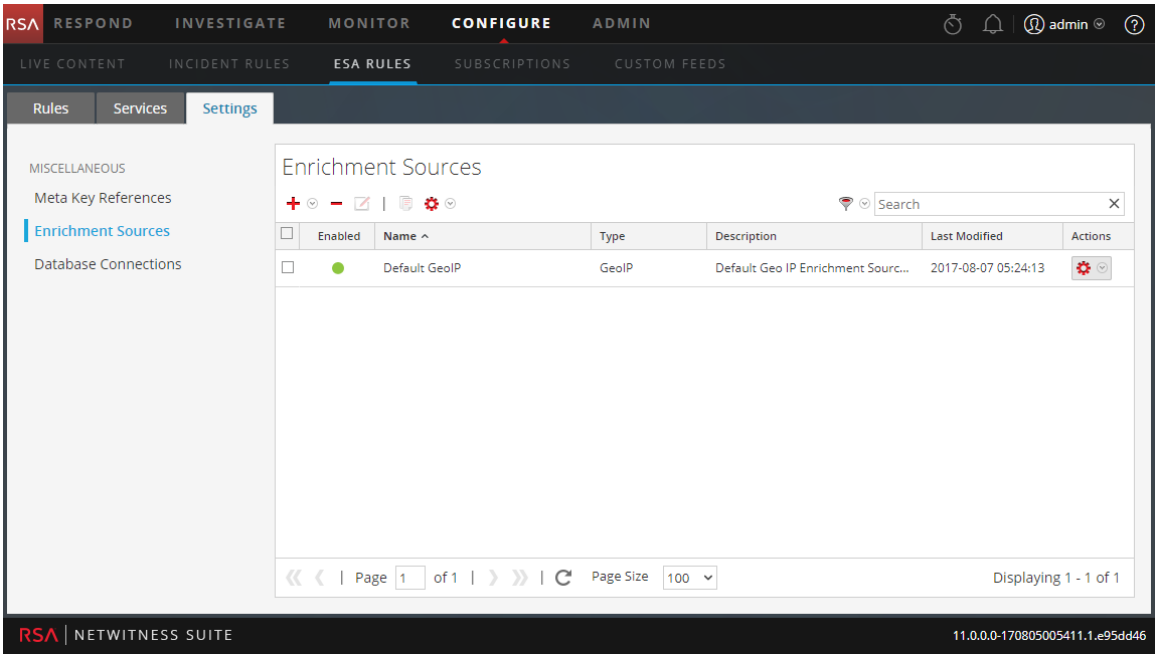
Configure Warehouse Analytics as an Enrichment Source

This topic provides instructions on how to configure RSA Warehouse Analytics as an enrichment source for ESA. Data analysts can leverage Warehouse Analytics data to analyze session and log data.

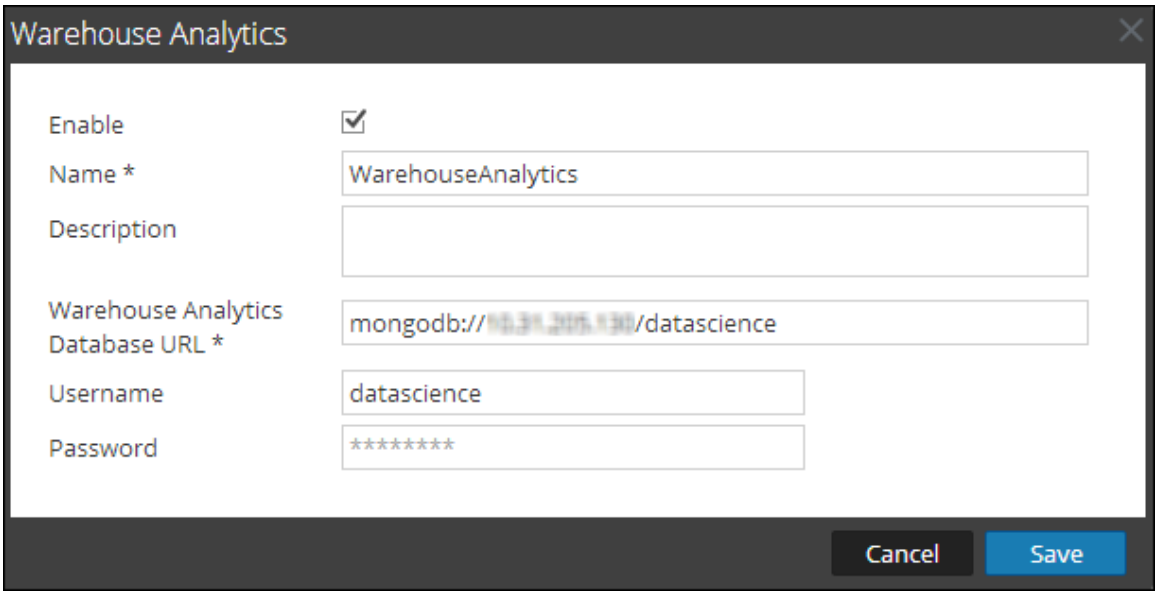
To configure Warehouse Analytics as an enrichment source:

1. Go to **CONFIGURE > ESA Rules > Settings** tab.
2. In the options panel, select **Enrichment Sources**.

   The Enrichment Sources panel is displayed.



3. From the + drop-down menu, select **Warehouse Analytics**.



4. Select **Enable** to enrich alerts with additional data. This is selected by default. If disabled, the alerts will not be enriched with additional data.
5. In the **Name** field, type a name to identify or label the Warehouse Analytics configuration.
6. In the **Description** field, type a brief description about the Warehouse Analytics configuration.
7. In the **Warehouse Analytics Database URL** field, type theMongoDb URL to the Warehouse Analytics database.
8. In the **Username** field, type the username to access the MongoDB.
9. In the **Password** field, type the password to access the MongoDB.
10. Click **Save**.

For more information, see Settings Tab.

Add an Enrichment to a Rule

This topic tells how to add a previously configured enrichment source to a rule. When ESA creates an alert, information from the source gets included in it.

Adding an enrichment to a rule allows you to request for look ups into a variety of sources and include the results in the outgoing alerts, giving you a more detailed alert. This procedure requires role permissions for Administrator, DPO, and SOC Manager.

Procedure

To add an enrichment to a rule:

1. Go to **CONFIGURE > ESA Rules**.
2. In the **Rule Library** view, do one of the following:
   - Double-click a rule.
   - Select a rule and click in the **Rule Library** toolbar.

   The Rule Builder panel is displayed in a new NetWitness Suite tab.
3. In the **Enrichments** section, click and select any of the following enrichment types:
   - In-Memory Table
   - External DB Reference
   - Warehouse Analytics
   - GeoIP

   Note: If you us a GeoIP source, ipv4 is automatically populated, and is not editable.

   The enrichment types that you have selected are displayed in the table.
4. For the added enrichment type, perform the following:
   - In the **Output** column, select the type that you have configured.
   - In the **Enrichment Source** drop-down list, select the enrichment source defined.
   - In the **ESA Event Stream Meta** field, type the event stream meta key whose value will be used as one operand of join condition.

   - In the **Enrichment Source Column Name** field, type the enrichment source column name whose value will be used as another operand of the join condition.
5. Select **Debug**. This will add a @Audit('stream') annotation to the rule. This is useful when debugging the esper rules.
6. Click **Show Syntax** to test if the defined ESA rule is valid.
7. Click **Save**.

For details on parameters and their descriptions, see Rule Builder Tab.

Deploy Rules to Run on ESA

This topic explains how to select an ESA and the rules to run on it. Administrator, SOC Manager or DPO role permissions are required for all tasks in this section.

To create a deployment, you need to perform the steps described in Deployment Steps

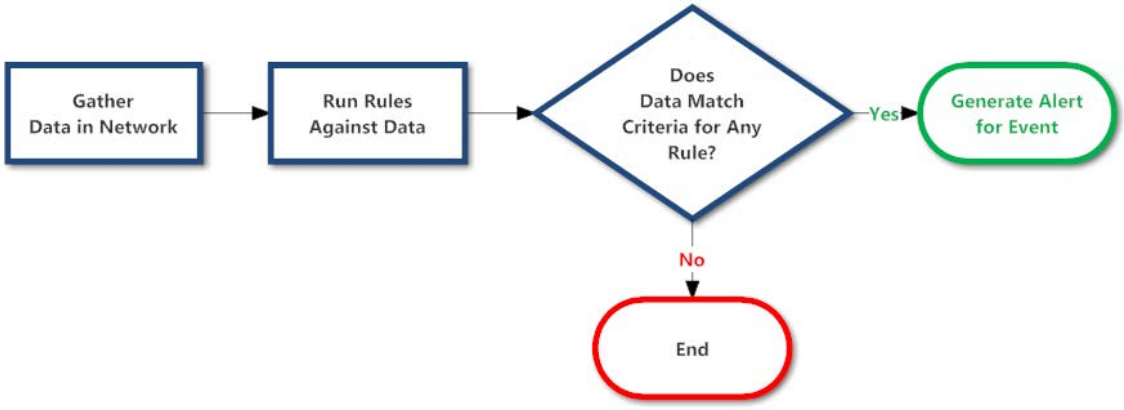
How Deployment Works

A deployment consists of an ESA service and a set of ESA rules. When you deploy rules, the ESA service runs them to detect suspicious or undesirable activity in your network. Each ESA rule detects a different event, such as when a user account is created and deleted within one hour.

The ESA service performs the following functions:

1. Gathers **data** in your network
2. Runs ESA **rules** against the data
3. Applies rule **criteria** to data
4. Generates an **alert** for the captured event

The following graphic shows this workflow:



In addition, you may want to perform other steps on your deployment, such as deleting an ESA service in your deployment, editing or deleting a rule from your deployment, editing or deleting a deployment, or showing updates to a deployment. For descriptions of these procedures, see Additional Deployment Procedures

Deployment Steps

This topic explains how to add a deployment, which includes an ESA service and a set of ESA rules. You can add a deployment to organize and manage ESA services and rules. Think of the deployment as a container for both components:

1. An ESA service
2. A set of ESA rules

For example, if you add a Spam Activity deployment it could include ESA London and a set of ESA rules to detect suspicious email activity.

To add a deployment, you need to complete the following procedures:

- Step 1. Add a Deployment
- Step 2. Add an ESA Service
- Step 3. Add and Deploy Rules

Step 1. Add a Deployment

Prerequisites

The following are required to add a deployment:

- The ESA service must be configured on the host. See "Configure ESA" in the *Event Stream Analysis (ESA) Configuration Guide*.
- Rules must be in the Rule Library. See Add Rules to the Rule Library.

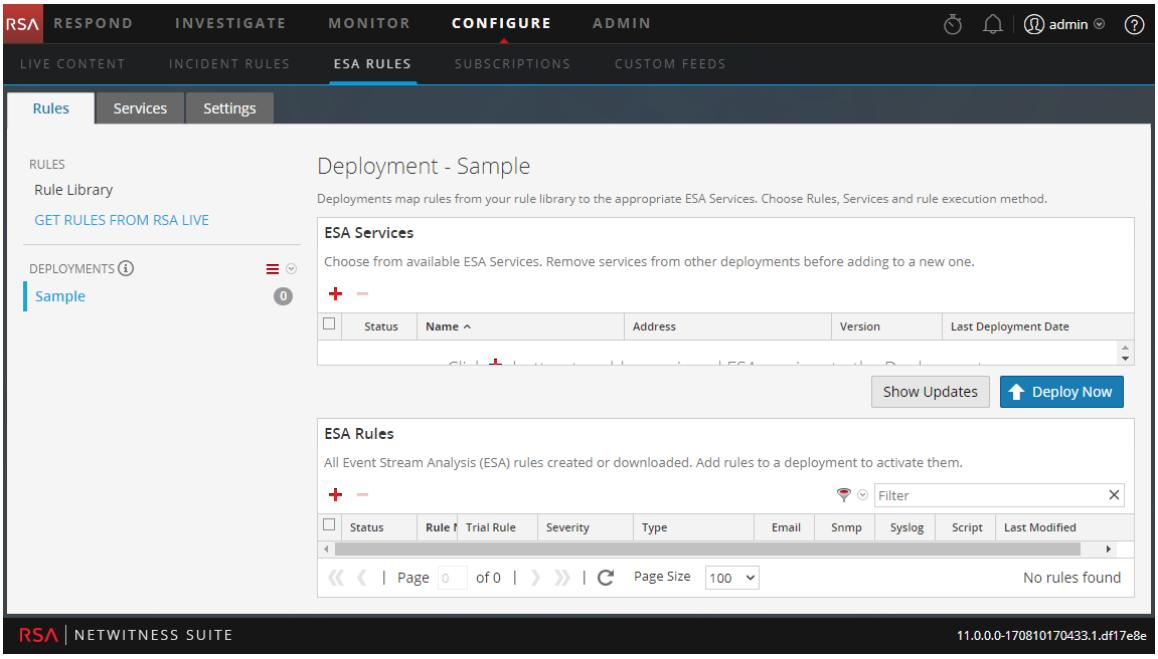
Procedure

To add a deployment:

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab is displayed.

2. In the options panel, next to Deployments, select  > **Add**.

   The Deployment view is displayed on the right.

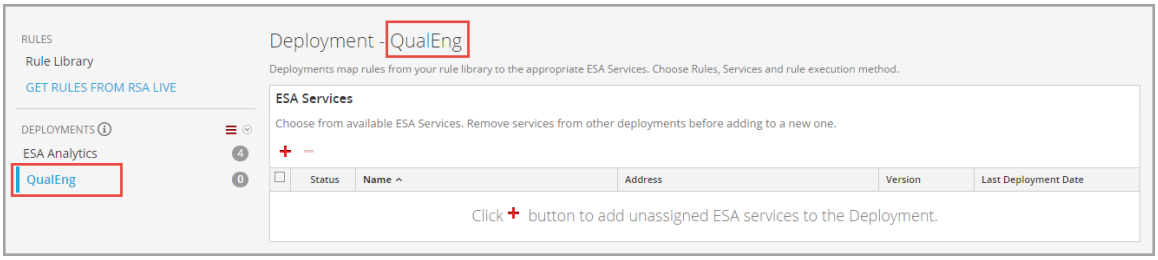


3. In the options panel, type a **name** for the deployment. The naming convention is up to you.

   For example, it could indicate the purpose or identify an owner.

4. Press **Enter**.

   The deployment is added.



Step 2. Add an ESA Service

The ESA service in a deployment gathers data in your network and runs ESA rules against the data. The goal is to capture events that match rule criteria, then generate an alert for the captured event.

You can add the same ESA to multiple deployments. For example, ESA London could be in the these deployments simultaneously:

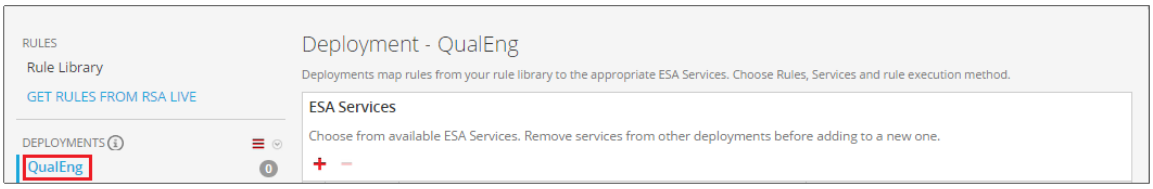
- Deployment EUR, which includes one set of ESA rules
- Deployment CORP, which includes another set of ESA rules

When you remove an ESA from a deployment, the rules are also removed from the ESA. For example, Deployment EUR could include ESA London and a set of 25 rules. If you remove ESA London from Deployment EUR, the 25 rules are also removed from ESA London. Consequently, if an ESA is not part of any deployment the ESA does not have any rules.

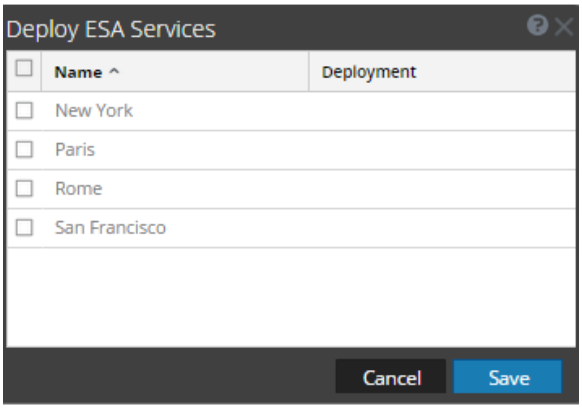
Procedure

To add an ESA service:

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab is displayed.
2. In the options panel, select a **deployment**:



3. In the **Deployment** view, click + in **ESA Services**.
   The Deploy ESA Services dialog lists each configured ESA.



4. Select an ESA and click **Save**.
   The Deployment view is displayed. The ESA is listed in the **ESA Services** section, with the status Added.

Step 3. Add and Deploy Rules

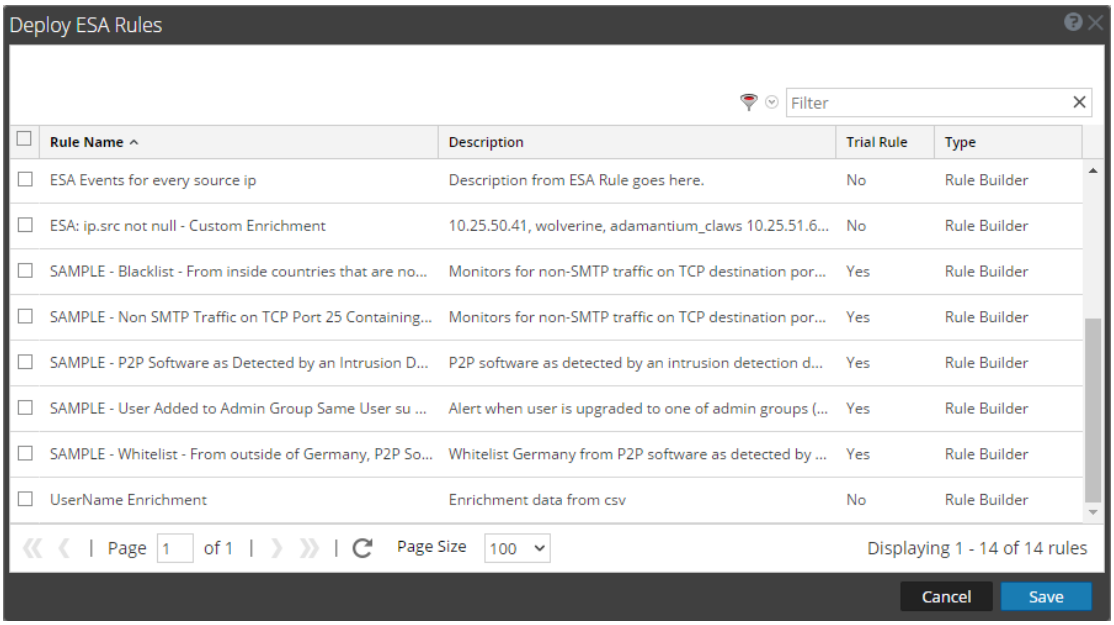
This topic explains how to add ESA rules to a deployment and then deploy the rules on ESA. Each ESA rule has unique criteria. The ESA rules in a deployment determine which events ESA captures, which in turn determine the alerts you receive.

For example, Deployment A includes ESA Paris and, among others, a rule to detect file transfer using a non-standard port. When ESA Paris detects a file transfer that matches the rule criteria, it captures the event and generates an alert for it. If you remove this rule from Deployment A, ESA will no longer generate an alert for such an occurrence.

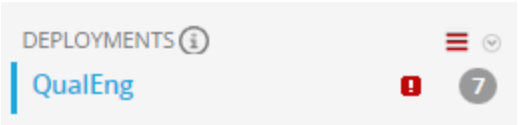
Procedure

To add and deploy rules:

1. Go to **Configure > ESA Rules**.
   The Rules tab is displayed.
2. In the options panel, select a deployment.
3. In the **Deployment** view, click + in **ESA Rules**.
   The Deploy ESA Rules dialog is displayed and shows each rule in your Rule Library:



4. Select rules and click **Save**.
   The Deployment view is displayed.
5. The rules are listed in the ESA Rules section.
   - In the Status column, **Added** is next to each new rule.
   - In the Deployments section, indicates there are updates to the deployment.
   - The total number of rules in the deployment is on the right.



6. Click **Deploy Now**.
   The ESA service runs the rule set.

Additional Deployment Procedures

In addition to deploying an ESA service and rules, you may want to perform other steps on your deployment, such as deleting an ESA service in your deployment, editing or deleting a rule from your deployment, editing or deleting a deployment, or showing updates to a deployment.

To perform these procedures, go to:

- Delete ESA Service in a Deployment
- Edit or Delete Rule in a Deployment
- Edit or Delete a Deployment
- Show Updates to a Deployment

Delete ESA Service in a Deployment

This topic provides instructions to delete an ESA service in a deployment. On a deployment with a service, you can edit the rules which are applied to the service and delete the service from the deployment.

Each of the following procedures starts in the Rules tab.

Procedure

To delete an ESA service:

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
   The Rules tab is displayed.
2. In the options panel, under **Deployments**, select a deployment.
3. In the **ESA Services** panel, select a service and click in the toolbar.
   A confirmation dialog is displayed.
4. Click **Yes**.
   The service is deleted.

Edit or Delete Rule in a Deployment

On a deployment with rules, you can edit and delete rules to customize the deployment. Each of the following procedures starts in the Rules tab.

Procedures

Edit a Rule

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
   The Rules tab is displayed.
2. In the options panel, under Deployments, select a deployment.
3. In the **ESA Rules** panel, double-click a rule to open it in a new tab.
4. Modify the rule, then click **Apply**.
   The rule is saved.

Delete a Rule

1. Go to **CONFIGURE > ESA Rules > Rules** tab.
   The Rules tab is displayed.
2. In the options panel, under **Deployments**, select a deployment.
3. In the **ESA Rules** panel, select a rule and click in the toolbar.
   A confirmation dialog is displayed.
4. Click **Yes**.
   The rule is deleted.

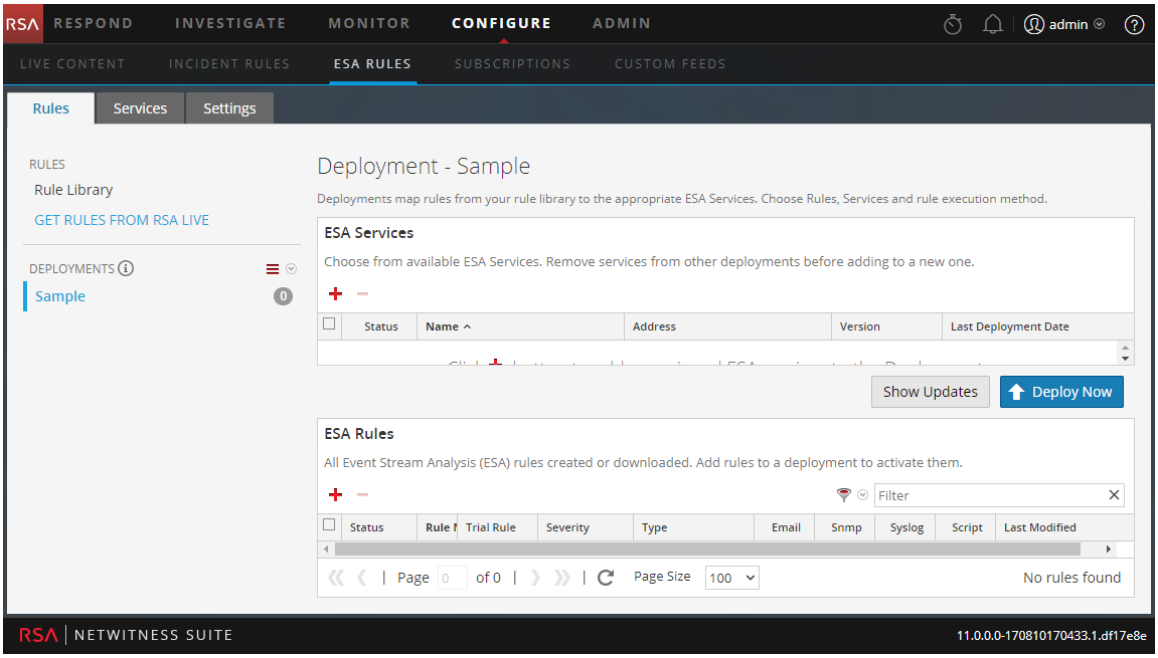
Edit or Delete a Deployment

This topic explains how NetWitness Suite forwards a correlation rule to each ESA service in a correlation group. In a correlation group, each ESA service must run the same set of rules. When you add a rule to a correlation group, NetWitness Suite forwards the rule to each ESA in the group.

To access the deployments:

1. Go to **CONFIGURE > ESA Rules**.
   The Configure view is displayed with the Rules tab open.
2. In the options panel, under **Deployments**, select a deployment.

The Deployment view is displayed.



Edit a Deployment

1. In the options panel, under **Deployments**, select a deployment.

   The Deployment view is displayed.

2. Select > **Edit**.

   The deployment name is made available for editing.

Delete a Deployment

1. In the options panel, under **Deployments**, select a deployment.

   The Deployment view is displayed.

2. Select > **Delete**.

   A confirmation dialog is displayed.

3. Click **Yes**.

   The deployment is deleted.

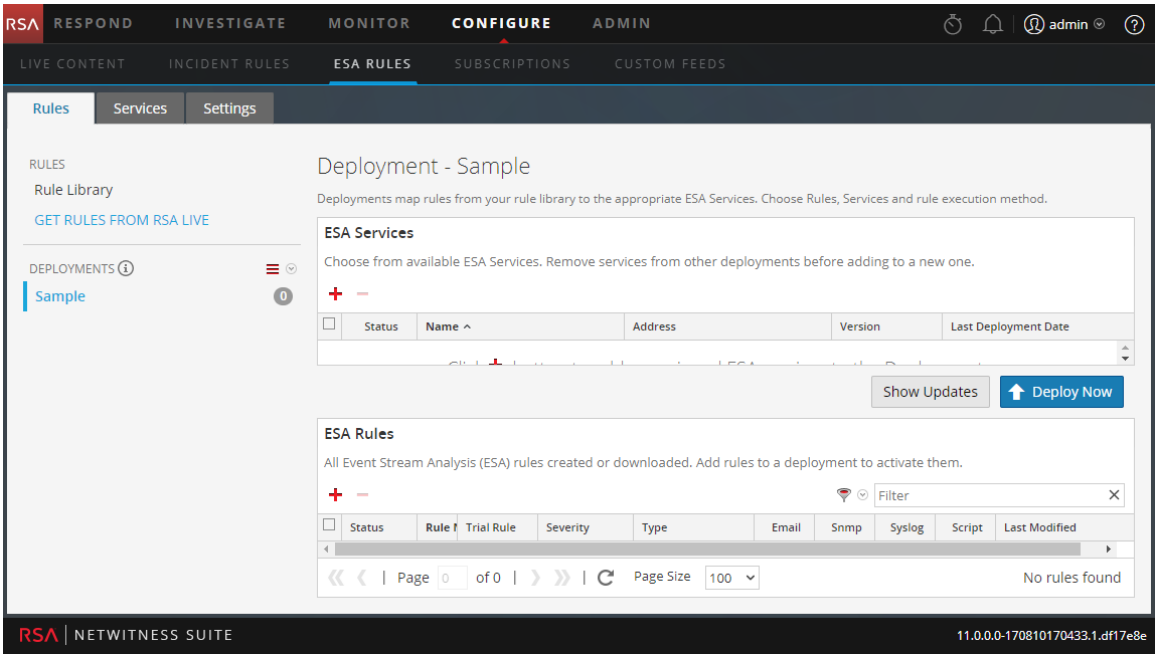
Show Updates to a Deployment

This topic explains how to show updates, such as adding or deleting rules, to a deployment. When you make a change to a deployment, the update icon ( ) appears next to the name of the deployment.

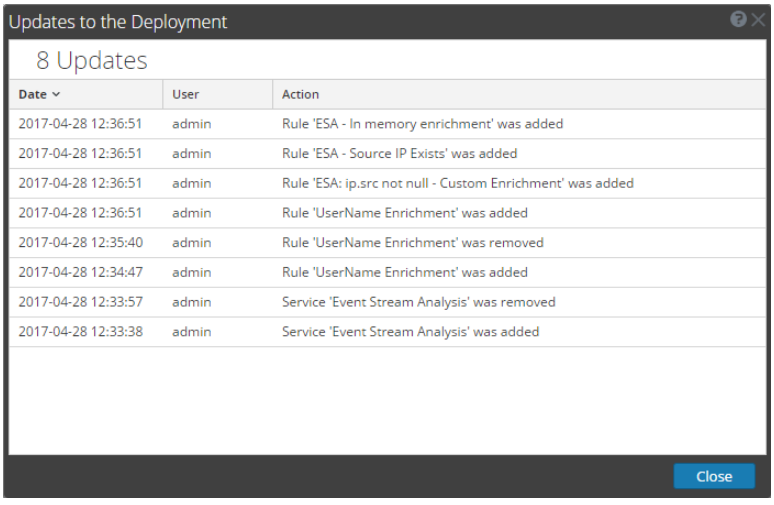
Procedure

To show the updates to a deployment:

1. Go to **CONFIGURE > ESA Rules.**
   The Rules tab is displayed.
2. In the options panel, under **Deployments** click **Show Updates** on the far right.



   The Updates to the Deployments dialog opens and shows the changes to the deployment.



3. Click **Close**.

View ESA Stats and Alerts

When the ESA generates alerts, you can view details about how the rules performed, such as statistics on the engine, rule, and alert, and you can also view information on which rules are enabled or disabled. For instructions on viewing ESA stats, see View Stats for ESA Service

When your ESA generates alerts, you can view the results in the Alerts Summary page. This enables you to see trends and understand both the volume and frequency of alerts. For instructions on viewing alerts, see View a Summary of Alerts

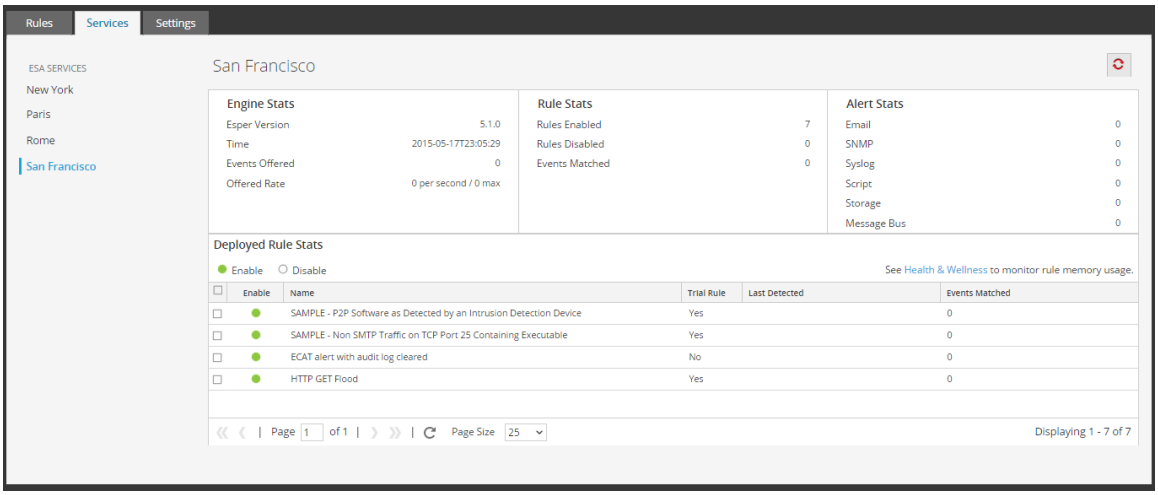
View Stats for ESA Service

This topic describes how to view the deployment stats for an ESA service. This procedure is useful when you are attempting to determine the effectiveness of a rule or troubleshoot a deployment.

Procedures

View ESA Stats

1. Go to **CONFIGURE > ESA Rules > Services** tab.
2. From the **ESA Services** list on the left, select a service.
   The deployment stats for the selected service are displayed.



3. Review the following sections of ESA stats.
   For a complete description of each statistic in each section, see Services Tab.

- **Engine Stats**
- **Rule Stats**
- **Alert Stats**

4. In the Deployed Rule Stats, review details about the rules deployed on the ESA.
   For a complete description of each column in each section, see Services Tab.
   - If the rule is enabled or disabled
   - What the rule name is
   - If the rule is running in Trial Rule mode
   - Last detected
   - Events matched
5. To get a snapshot of the rule memory, click **Health & Wellness**.

Enable or Disable Rules

1. In the **Deployed Rule Stats** panel, select a rule from the grid.
2. Click Enable to enable the rule, or click Disable to disable the rule.
   The Services tab is refreshed to show the changes, which take effect immediately.

Refresh the Statistics

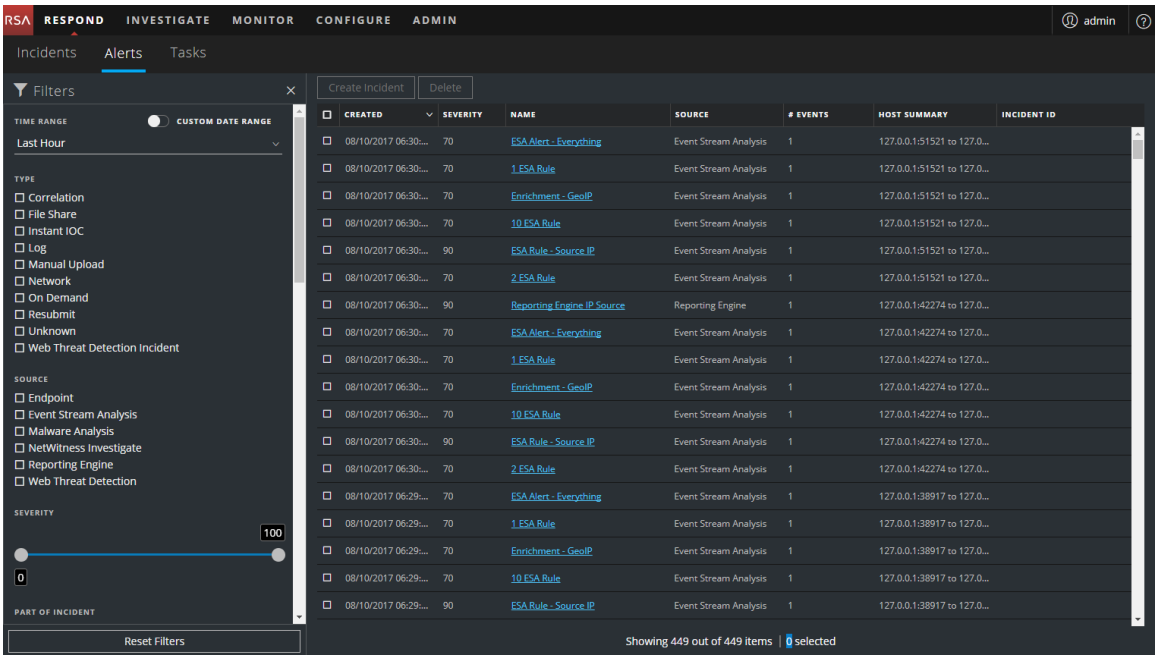
The Services tab does not update statistics automatically unless you enable or disable a rule. To ensure you view current statistics:

1. Click in the upper right corner to refresh the information.
2. View the updated information.

View a Summary of Alerts

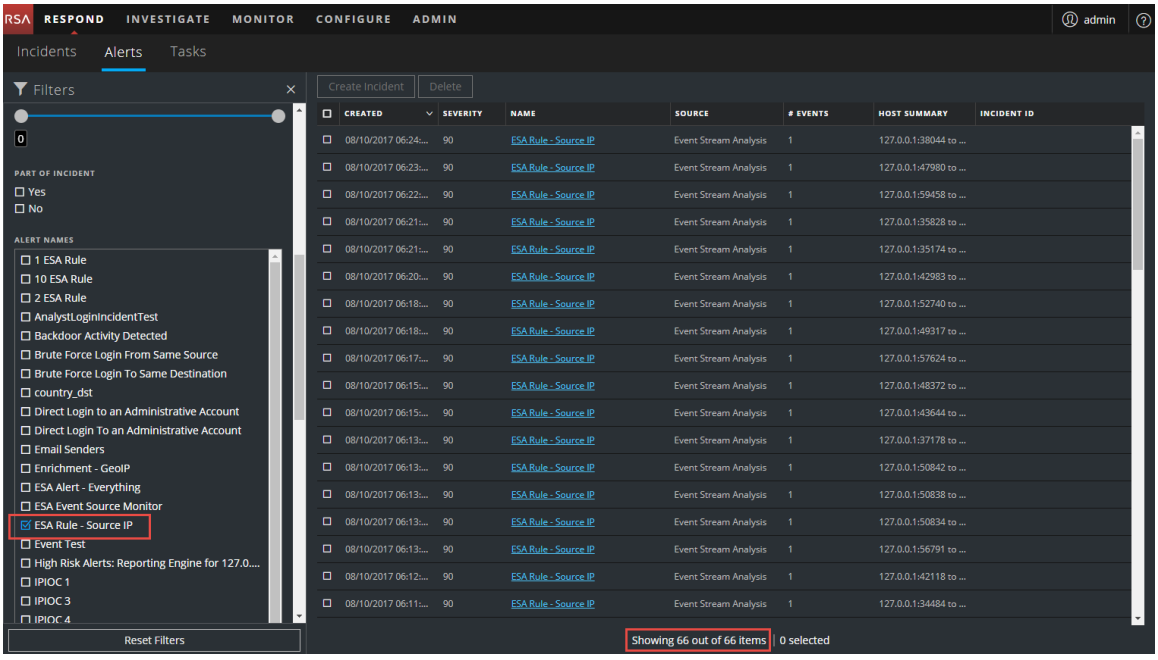
In the RESPOND view, you can browse through various alerts from multiple sources. You can filter the alerts list to show only alerts of interest, such as by Alert Name, alert source, and a specific time range.

1. Go to **RESPOND > Alerts**.
   The Respond Alerts List view displays a list of all NetWitness Suite alerts.



2. In the **Filters** panel on the left, you can filter the alerts list to view specific alerts for a specific time frame. For example, in the ALERT NAMES section, you can select an alert for an ESA rule, such as ESA Rule - Source IP, and leave the TIME FRAME set to Last Hour.

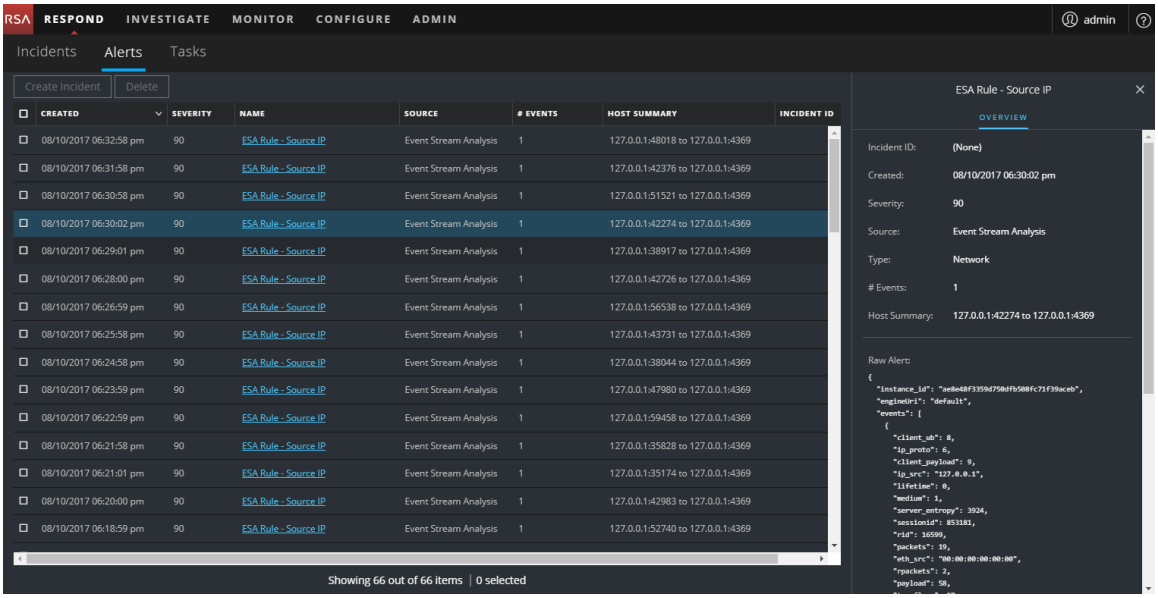
   The alerts list to the right shows a list of alerts that match your filter selection along with a count of the alerts at the bottom of the alerts list.



   The alerts list shows information about each of the alerts.

- **Created**: Displays the date and time when the alert was created in the source system.
- **Severity**: Displays the level of severity of the alert. The values are from 1 to 100.
- **Name**: Displays a basic description of the alert.
- **Source**: Displays the original source of the alert.
- **# of Events**: Indicates the number of events contained within an alert.
- **Host Summary**: Displays details of the host, like the host name from where the alert was triggered.
- **Incident ID**: Shows the incident ID of the alert. If there is no incident ID, the alert does not belong to an incident.

3. You can click on an alert in the list to open an **Overview** panel on the right where you can view raw alert metadata.



For more information about filtering alerts and viewing alert details, see the *NetWitness Respond User Guide*.

ESA Alert References

In the Alerts module, you configure and deploy ESA rules to get alerted about potential network threats.

These topics explain the user interface in the Alerts module.

- New Advanced EPL Rule Tab
- Build a Statement Dialog
- Deploy ESA Rules Dialog
- Deploy ESA Services Dialog
- Rule Builder Tab
- Rules Tab
- Rule Syntax Dialog
- Services Tab
- Settings Tab
- Updates to the Deployment Dialog

New Advanced EPL Rule Tab

The Advanced EPL Rule tab enables you to define rule criteria with an Event Processing Language (EPL) query.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Define an Advanced EPL rule.	Add an Advanced EPL Rule
Content Expert	See examples of an Advanced EPL Rule.	Sample Advanced EPL Rules

Related Topics

- Add a Rule Builder Rule
- Enrichment Sources

Advanced EPL Rule

To access the Advanced EPL Rule tab:

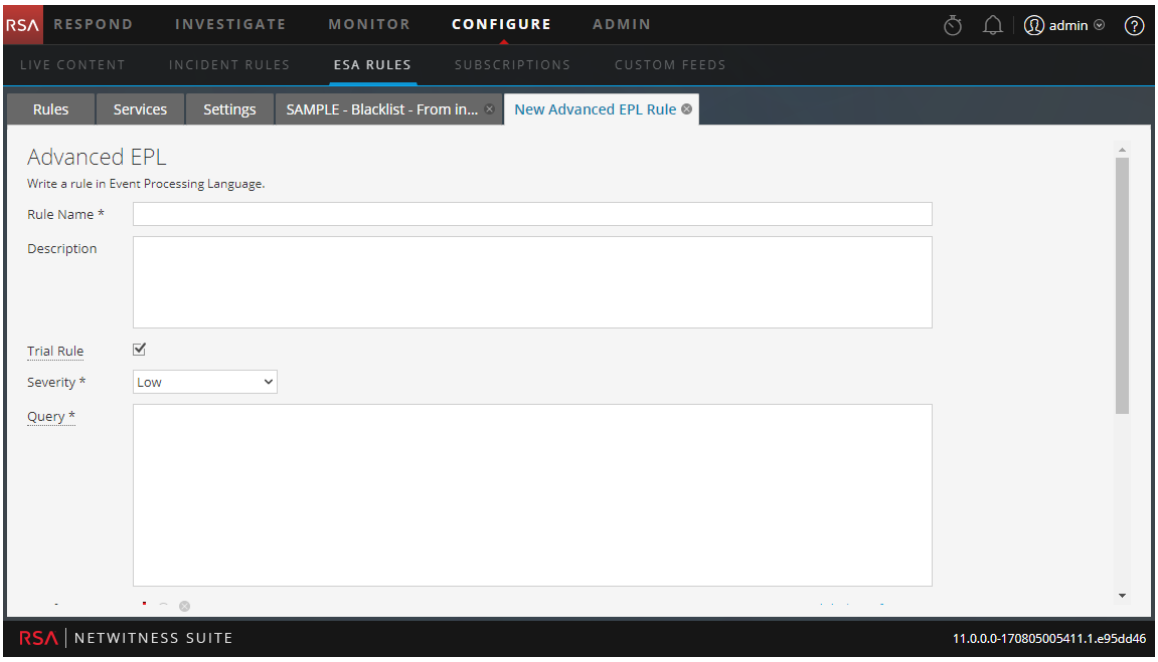
1. Go to **CONFIGURE > ESA Rules**.

   The Configure view is displayed with the Rules tab open by default.

2. In the **Rule Library** toolbar, select  **> Advanced EPL**.

   The Advanced EPL Rule tab is displayed.

Below is a screen shot of the Advanced EPL Rule tab.



The following table lists the parameters in the Advanced EPL Rule tab.

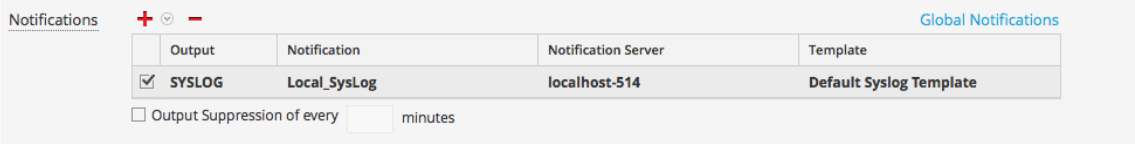
Parameters	Description
Rule Name	Purpose of the ESA rule.
Description	Summary of what the ESA rule detects.
Trial Rule	Deployment mode to see if the rule runs efficiently.
Severity	Threat level of alert triggered by the rule.
Query	EPL query that defines rule criteria.

Notifications

In the Notifications section, you can choose how to be notified when ESA generates an alert for the rule.

For more information on the alert notifications, see Add Notification Method to a Rule.

The following figure shows the Notifications section.



Parameter	Description
+	To add an alert notification type.
−	To delete the selected alert notification type.
Output	Alert notification type. Options are: • Email • SNMP • Syslog • Script
Notification	Name of previously configured output, such as an email distribution list.
Notification Server	Name of server that sends the output.
Template	Name of template for the alert notification.
Output Suppression of every	Option to specify alert frequency.
Minutes	Alert frequency in minutes.

Enrichments

In the Enrichments section, you can add a data enrichment source to a rule.

For more information on the enrichments, see Add an Enrichment to a Rule.
The following figure shows the Enrichments section.

Parameter	Description
+	To add an enrichment.

Parameter	Description
	To delete the selected enrichment.
Output	Enrichment source type. Options are: • In-Memory Table • External DB Reference • Warehouse Analytics • GeoIP
Enrichment Source	Name of previously configured enrichment source, such as a .CSV filename for an In-Memory Table.
ESA Event Stream Meta	ESA meta key whose value will be used as one operand of join condition.
Enrichment Source Column Name	Enrichment source column name whose value will be used as the other operand of the join condition.

Build a Statement Dialog

The Build a Statement dialog allows you to construct a condition statement when creating a new Rule Builder rule.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Configure a rule statement.	Add an Advanced EPL Rule
Content Expert	Add conditions to the rule.	Step 3. Add Conditions to a Rule Statement

Related Topics

- Add a Rule Builder Rule

Build a Statement Dialog

To access the Build a Statement dialog:

1. Go to **CONFIGURE > ESA Rules**.

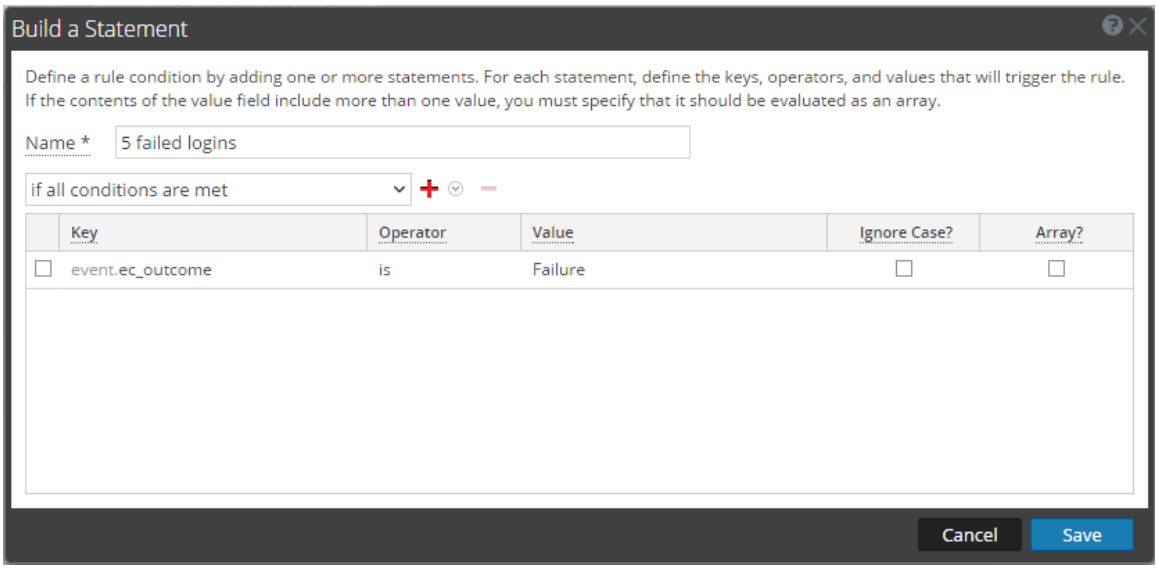
   The Configure ESA Rules view is displayed with the Rules tab open.

2. In the **Rule Library** toolbar, select  **> Rule Builder**.

   A New Rule tab is displayed..

3. In the **Conditions** section, click  .

   The Build a Statement dialog is displayed.



The following table describes the parameters in the Build a Statement dialog.

Parameter	Description
Name	Purpose of the statement.
Select	Conditions the rule requires. There are two options: • If all conditions are met • If any of these conditions are met
Key	Key for ESA to check in the rule statement.

Parameter	Description
Evaluation Type	Relationship between the meta key and value for the key: • is • is not • is not null • is greater than (>) • is greater than or equal to (>=) • is less than (<) • is less than or equal to (<=) • contains • not contains • begins with • ends with
Value	Value for ESA to look for in the key.
Ignore Case?	This field is designed for use with string and array of string values. By choosing the **Ignore Case** field, the query will treat all string text as a lowercase value. This ensures that a rule that searches for the user named Johnson would trigger if the event contains "johnson," "JOHNSON," or "JoHnSoN."
Array?	Choice to indicate if contents of Value field represent one value or multiple values: • Select the box to indicate multiple values. • Clear the box to indicate one value.
+	Add a statement. You can add a meta condition, whitelist condition, or blacklist condition.
−	Delete selected statement.
Save	Add statement to the Conditions section of the Rule Builder tab.

The following table shows the operators you can use in the Rule Builder:

Operator	Required Value	Usage	Example	Meaning
is	Singular string value	The meta key is equal to the *value* field.	***user_dst*** is John Doe.	***user_dst*** is equal to the string *"John Doe"*.
is	Array string value	The meta key is equal to one of the elements of the *value* field.	*user_dst* is John, Doe, Smith.	*user_dst* is equal either to the string *"John"* or to the string *"Doe"* or to the string *"Smith"* (Note, the spaces are stripped.).
is not	Singular string value	The meta key is not equal to the *value* field.	***size*** is not 200.	***size*** is not equal to the number 200 (size is a numeric value).
is not	Array string value	The meta key is not equal to any of the elements of the *value* field.	***size*** is not 200, 300, 400.	***size*** is equal neither to *200* nor to *300* nor to *400*.
is not null	N/A (looks for any value)	The meta key value is not null.	***user_dst*** is not null.	***user_dst*** is a meta that contains a value.
is greater than (>)	Number	The numeric value of the meta key is greater than the number in the *value* field.	***payload*** is greater than 7000.	***payload*** is a numeric value that is greater than 7000.
is greater than or equal to (>=)	Number	The numeric value of the meta key is greater than or equal to the number in the *value* field.	***payload*** is greater than or equal to 7000.	***payload*** is a numeric value that is greater than or equal to 7000.
is less than (<)	Number	The numeric value of the meta key is less than the number in the *value* field.	***ip_dstport*** is less than 1024.	***ip_dstport*** is a numeric value that is less than the numeric value 1024.
is less than or equal to (<=)	Number	The numeric value of the meta key is less than or equal to the number in the *value* field.	***ip_dstport*** is less than or equal to 1024.	***ip_dstport*** is a numeric value that is less than or equal to numeric value 1024.
contains	String	The *value* field is a substring of the meta key (This operator is only available for a string-valued meta key).	***ec_outcome*** contains failure.	***ec_outcome*** is a string that contains the substring *"failure"*.
not contains	String	The *value* field is not a substring of the meta key (This operator is only available for a string-valued meta key).	***ec_outcome*** not contains failure.	***ec_outcome*** is a string that does not contain the substring*"failure"*.

Operator	Required Value	Usage	Example	Meaning
begins with	String	The *value* field is the beginning of the meta key (This operator is only available for a string-valued meta key).	***ip_dst*** begins with 127.0.	***ip_dst*** is a string that starts with *"127.0"*.
ends with	String	The *value* field is the end of the meta key (This operator is only available for a string-valued meta key).	***user_dst*** ends with son.	***user_dst*** is a string that ends in*"son"*.

Note: Terms in ***bold italic*** are Meta that may not exist in all customer enviroments.

Deploy ESA Rules Dialog

The Deploy ESA Rules dialog enables you to filter and select rules to deploy to an ESA service.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Configure a deployment.	Step 1. Add a Deployment
Content Expert	Deploy a rule	Step 3. Add and Deploy Rules

Related Topics

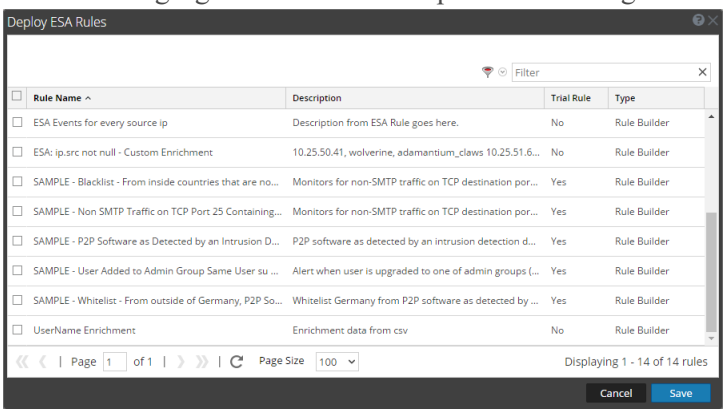
- Additional Deployment Procedures

Deploy ESA Rules Dialog

To access this dialog:

1. Go to **CONFIGURE** > **ESA Rules**.
   The Rules tab opens by default.
2. In the options panel, under the **Deployment** section, select or add a new deployment by clicking  > **Add**.
3. If you add a new deployment, type the name of the deployment in the box in the options panel.
4. In the **ESA Rules** panel, click  .
   The Deploy ESA Rules dialog is displayed.

The following figure shows an example of this dialog.



The following table describes the parameters of the Deploy ESA Rules dialog.

Parameters	Description
	Filters the list of rules based on severity and type. The text box beside this icon filters based on rule name.
Rule Name	Displays the name of the rule.
Description	Describes the rule.
Trial Rule	Indicates whether or not the rule is a trial rule.
Type	Indicates the type of rule: RSA Live ESA, Advanced EPL, or Rule Builder.

Deploy ESA Services Dialog

The Deploy ESA Services dialog displays all ESA services available to be added to a deployment.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Configure a deployment.	Step 1. Add a Deployment
Content Expert	Deploy a service	Step 2. Add an ESA Service

Related Topics

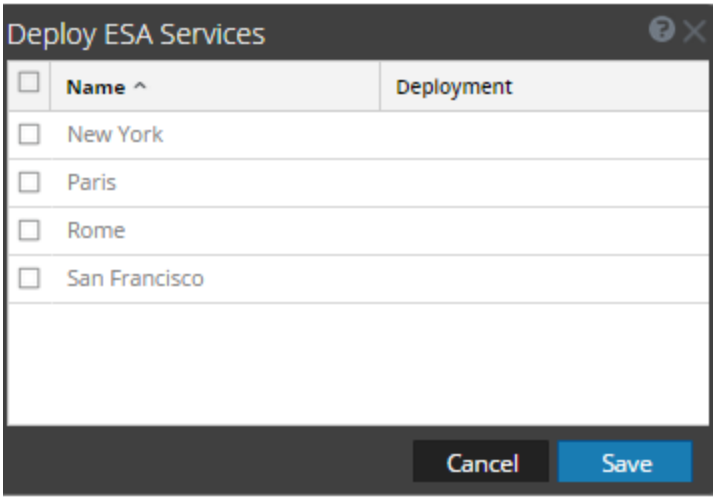
- Additional Deployment Procedures
- View Stats for ESA Service

Deploy ESA Services Dialog

To access this dialog:

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab opens by default.
2. In the options panel, under the **Deployment** section, select or add a deployment.
3. In the **ESA Services** panel, click  .
   The Deploy ESA Services dialog is displayed.

The following figure is an example of this dialog.



The following table describes the parameters of the Deploy ESA Services dialog.

Parameters	Description
Name	Displays the name of configured ESA services.
Deployment	Displays the deployments to which the service has already been added.

Rule Builder Tab

The Rule Builder tab enables you to define a Rule Builder rule.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Define a Rule Builder rule.	Add a Rule Builder Rule
Content Expert	Define rule criteria.	Step 2. Build a Rule Statement
Content Expert	Add conditions to the rule.	Step 3. Add Conditions to a Rule Statement

Related Topics

- Add an Advanced EPL Rule

Rule Builder

To access the Rule Builder tab:

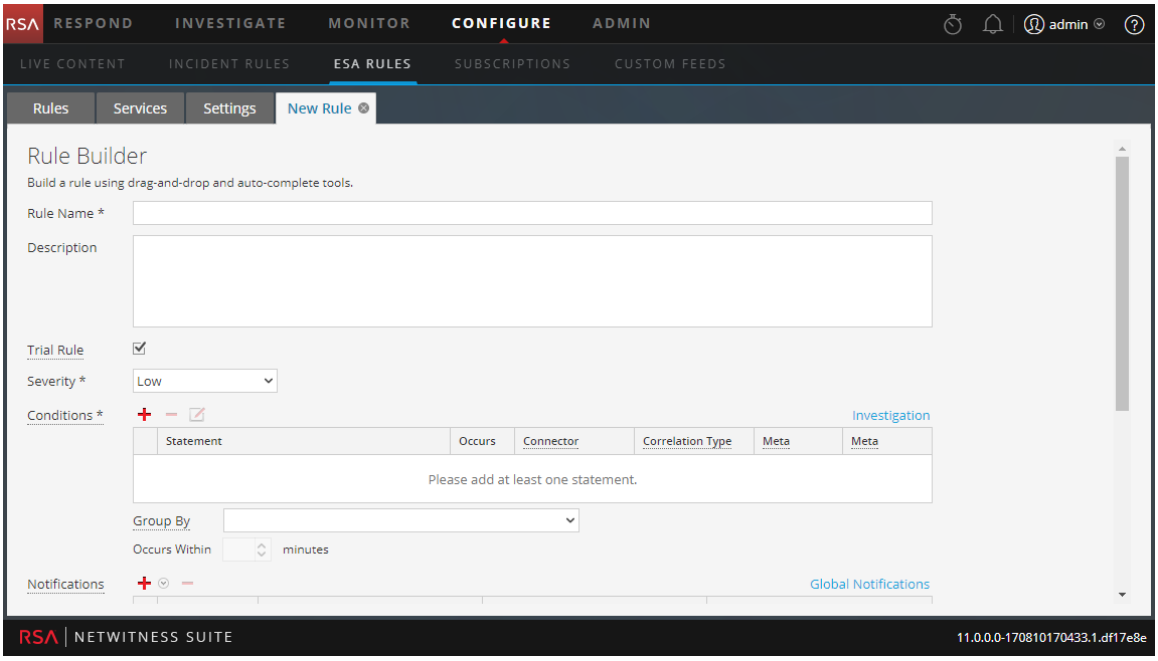
1. Go to **CONFIGURE** > **ESA Rules**.

   The Rules tab opens by default.

2. In the **Rule Library** toolbar, select  > **Rule Builder**.

   The Rule Builder tab is displayed.

The following figure shows the Rule Builder tab.



The following table lists the parameters in the Rule Builder tab.

Parameters	Description
Rule Name	Purpose of the ESA rule.
Description	Summary of what the ESA rule detects.
Trial Rule	Deployment mode to see if the rule runs efficiently.
Severity	Threat level of alert triggered by the rule.

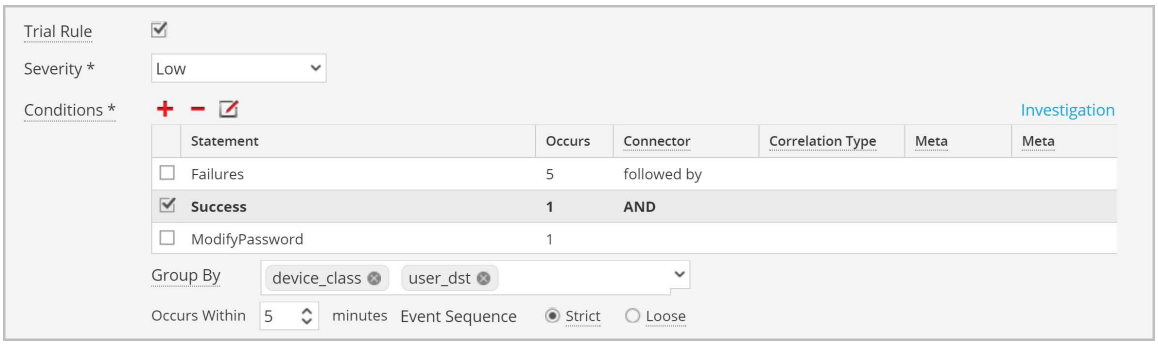
The Rule Builder includes the following components:

- Conditions section
- Notifications section
- Enrichments section

Conditions Section

In the Conditions section of the Rule Builder tab, you define what the rule detects.

The following figure shows the Conditions section.



The following table lists the parameters of the Conditions section.

Parameter	Description
+	Add a statement.
−	Remove selected statement.
✎	Edit selected statement.
Statement	Logical group of conditions for one operation.
Occurs	Alert frequency if the condition is met. This specifies that there must be at least that many events that satisfy the criteria in order to trigger an alert. The time window in minutes binds the Occurs count.

Parameter	Description
Connector	Options to specify relationship among the statements: • followed by • not followed by • AND • OR The Connector joins two statements with AND, OR, followed by, or not followed by. When followed by is used, it specifies that there is a sequencing of those events. AND and OR build one large criteria. The followed by creates distinct criteria that occurs in sequence.
Correlation Type	**Correlation Type** applies only to **followed by** and **not followed by**. If you choose a correlation type of SAME, select one meta to correlate on, and if you choose a correlation type of JOIN, select two meta to correlate on. You may want to use JOIN if you are trying to correlate on meta from two different data sources. For example, say you want to correlate an AV alert with an IDS alert.
Meta	Enter the meta condition if choosing a correlation type of SAME or JOIN (as described above).
Meta	Enter the second meta condition if choosing a correlation type of JOIN (as described above). For example, The destination IP address from the AV alert and source IP address for the workstation from the IDS alert are joined to allow you to view the same entities across different sources.
occurs within minutes	Time window within which the conditions must occur.

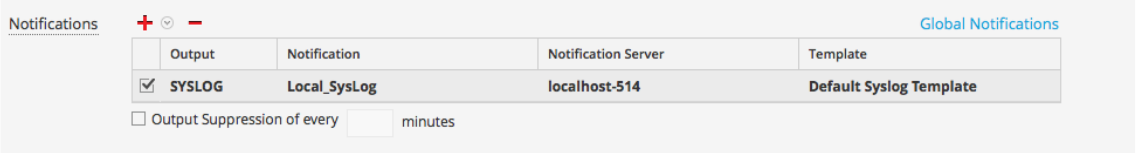
Parameter	Description
Event Sequence	Choose whether the pattern must follow a *strict* match or a *loose* match. If you specify a strict match, this means that the pattern must occur in the *exact* sequence you specified with no additional events occurring in between. For example, if the sequence specifies five failed logins (F) followed by a successful login (S), this pattern will only match if the user executes the following sequence: F,F,F,F,F,S. If you specify a loose match, this means that other events may occur within the sequence, but the rule will still trigger if all of the specified events also occur. For example, five failed login attempts (F), followed by any number of intervening successful login attempts (S), followed by a successful login attempt might create the following pattern: F,S,F,S,F,S,F,S,F,S which would trigger the rule despite the intervening successful logins.
Group By	Select the meta key by which to group results from the dropdown list. For example, suppose that there are three users; Joe, Jane, and John and you use the Group By meta, **user_dst** (user_dst is the meta field for the user destination account). The result will show events grouped under the user destination accounts, Joe, Jane, and John. You can also group by multiple keys. For example, you might want to group by user and machine to see if a user logged in from the same machine attempts to log into an account multiple times. To do this, you might group by device_class and user_dst.

Notifications

In the Notifications section, you can choose how to be notified when ESA generates an alert for the rule.

For more information on the alert notifications, see Add Notification Method to a Rule.

The following figure shows the Notifications section.



Parameter	Description
+	To add an alert notification type.
−	To delete the selected alert notification.

Parameter	Description
Output	Alert notification type. Options are: • Email • SNMP • Syslog • Script
Notification	Name of previously configured output, such as an email distribution list.
Notification Server	Name of server that sends the output.
Template	Name of template for the alert notification.
Output Suppression of every	Option to specify alert frequency.
Minutes	Alert frequency in minutes.

Enrichments

In the Enrichments section, you can add a data enrichment source to a rule.

For more information on the enrichments, see Add an Enrichment to a Rule.

The following figure shows the Enrichments section.

Parameter	Description
+	To add an enrichment.
−	To delete the selected enrichment.

Parameter	Description
Output	Enrichment source type. Options are: • In-Memory Table • External DB Reference • Warehouse Analytics • GeoIP
Enrichment Source	Name of previously configured enrichment source, such as a .CSV filename for an In-Memory Table.
ESA Event Stream Meta	ESA meta key whose value will be used as one operand of join condition.
Enrichment Source Column Name	Enrichment source column name whose value will be used as the other operand of the join condition. For an in-memory table, If you configured a key when creating a .CSV-based enrichment, this column automatically populates with the selected key. However, you can change it if you like. For a GeoIP enrichment source, ipv4 is automatically selected.

Rules Tab

The Rules tab enables you use to manage ESA rules and deployments.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	View types of rules.	ESA Rule Types
Content Expert	Deploy Trial Rules.	Work with Trial Rules
Content Expert	Create a rule.	Add Rules to the Rule Library
Content Expert	Deploy a rule.	Deploy Rules to Run on ESA

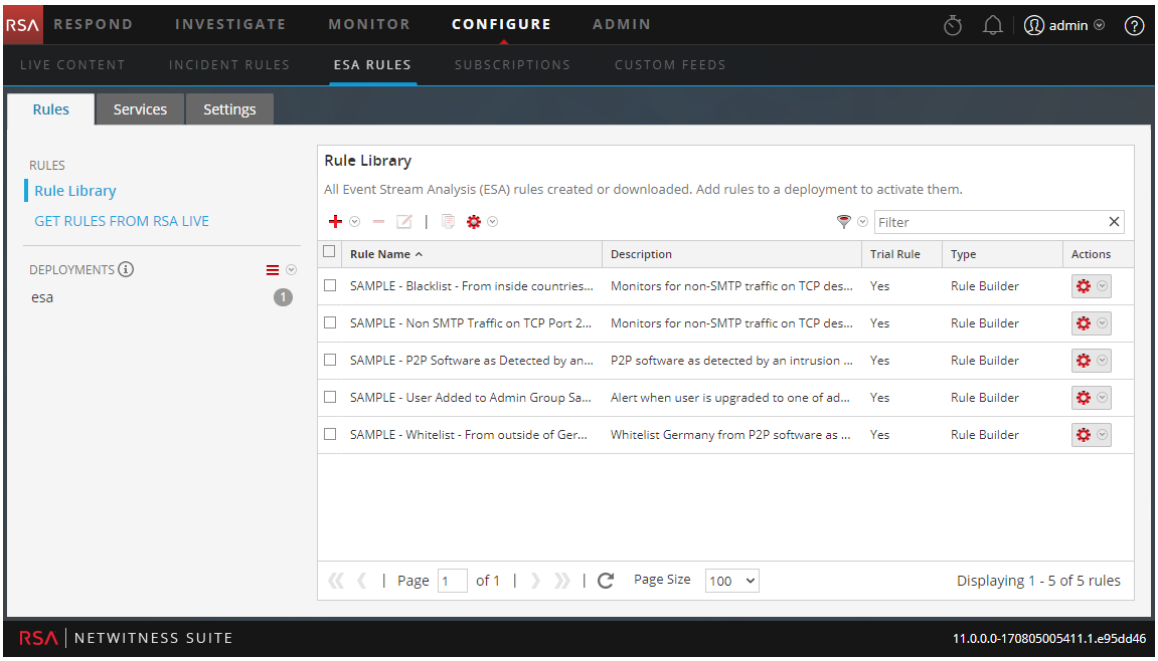
Related Topics

- Getting Started with ESA

Rule Builder

The Rules tab is displayed when you go to **CONFIGURE > ESA Rules**.

The following figure shows the Rules tab.



The Rules tab is divided into three sections:

- Rules Tab Options Panel
- Rule Library Panel
- Deployment Panel

Rules Tab Options Panel

In the **Rules** tab options panel to the left, you can view ESA rules in the Rule Library and create deployments.

What do you want to do?

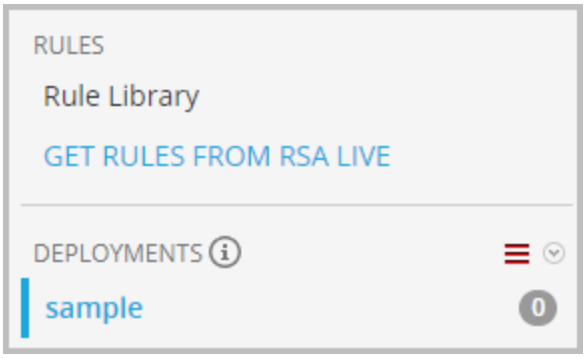
Role	I want to ...	Show me how
Content Expert	View an ESA rule.	Add Rules to the Rule Library
Content Expert	Create a deployment.	Deployment Steps

Related Topics

- Working with Rules

Options Panel

The following figure shows the options panel in the **Rules** tab.



There are two sections in the options panel: Rules and Deployments.

Rules Section

The Rules section contains two options. **Rule Library** is selected by default, and when it's selected, the Rule Library view is displayed within the tab. **Get Rules From RSA Live** navigates to the Live Search view, where you can search for rules.

Deployments Section

The Deployments section lists deployments and indicates whether there are updates to the deployments. From this section, deployments can be added, deleted, edited, and refreshed. Selecting a deployment from the list displays the Deployment panel within the tab. The following table describes the features of this section.

Feature	Description
	Displays a drop-down menu from which you can choose to add, edit, or delete a deployment. You can also refresh the list of deployments to see if there are any new updates to the list.
	Indicates whether there are any updates to the deployment.
	Indicates the number of rules in the deployment.

Rule Library Panel

The Rule Library panel allows you to manage rules.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Add an ESA rule.	Add a Rule Builder Rule
Content Expert	Edit, duplicate, or delete an ESA rule.	Edit, Duplicate or Delete a Rule
Content Expert	Import or export ESA rules.	Import or Export Rules
Content Expert	Filter the ESA rules list.	Filter or Search for Rules

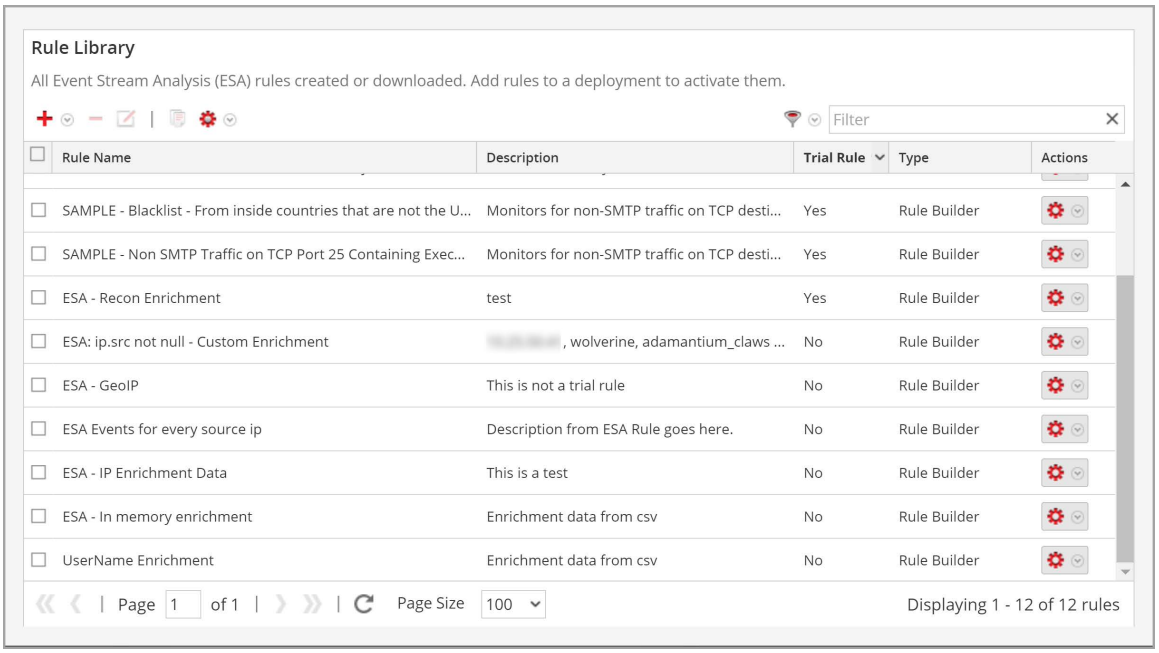
Related Topics

- Add an Advanced EPL Rule

Rule Library Panel

To access this view, go to **CONFIGURE > ESA Rules**. The Rules tab is displayed and the Rule Library panel is on the right.

The following figure shows the Rule Library panel.



The Rule Library panel includes the following components:

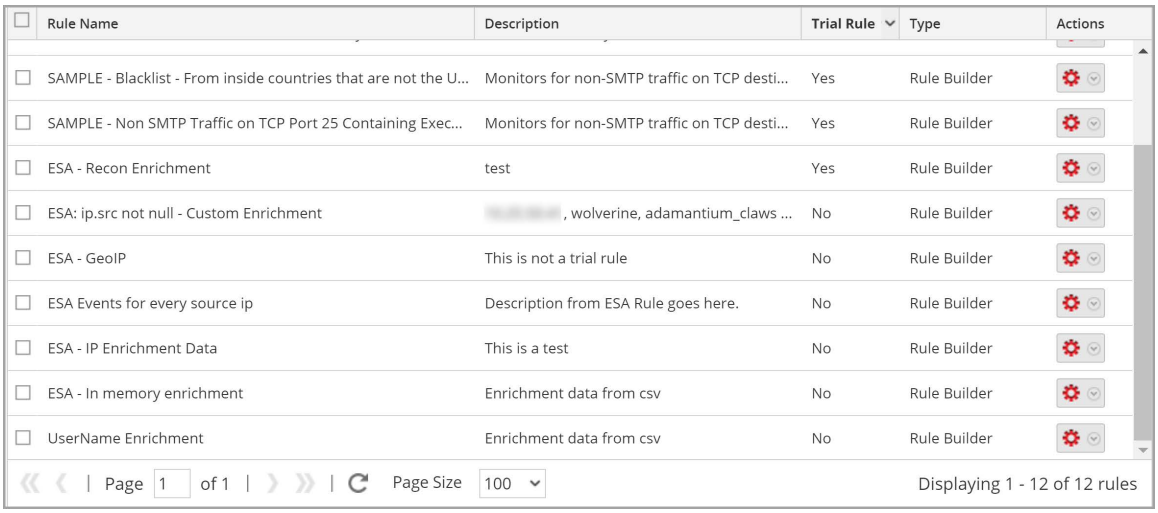
- Rule Library toolbar
- Rule Library list

Rule Library Toolbar

The Rule Library toolbar allows you to add, delete, edit, duplicate, filter, export, and import ESA rules. The following figure shows the icons for these actions.

Rule Library List

The following figure shows the Rule Library list.

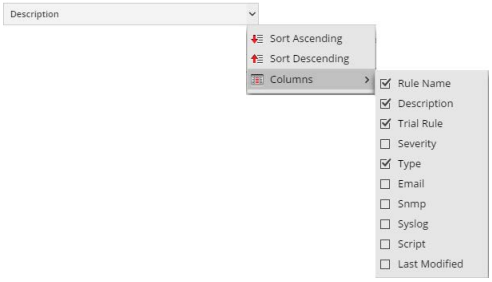


The Rule Library list shows all the ESA rules that have been downloaded from RSA Live or created in the Advanced EPL and Rule Builder tabs. The following table lists the columns in the Rule Library list and their description.

Column	Description
Rule Name	Purpose of the ESA rule.
Description	Summary of what the ESA rule detects.
Trial Rule	Deployment mode to see if the rule runs efficiently.
Type	The type of rule.

Column	Description
Actions ()	Menu to delete, edit, duplicate, or export the selected rule.
Severity	Threat level of alert triggered by the rule.
Email	Indicates whether an alert notification for the rule is sent by email. This column is not visible by default.
Snmp	Indicates whether an alert notification for the rule is sent using SNMP. This column is not visible by default.
Syslog	Indicates whether an alert notification for the rule is sent using Syslog. This column is not visible by default.
Script	Indicates whether an alert notification for the rule executes a script. This column is not visible by default.
Last Modified	The date and time when the ESA rule was last modified. This column is not visible by default.

To display columns which aren't visible by default, hover over the title of a column and click the **v** on the right. This opens a drop-down menu in which you can sort the contents of the column or choose which columns you want to see in the Rule Library list.



Deployment Panel

This topic provides an overview of the Deployment panel. The Deployment panel enables you to create and configure the deployments. The Deployment panel includes the following sections:

- ESA Services
- ESA Rules

What do you want to do?

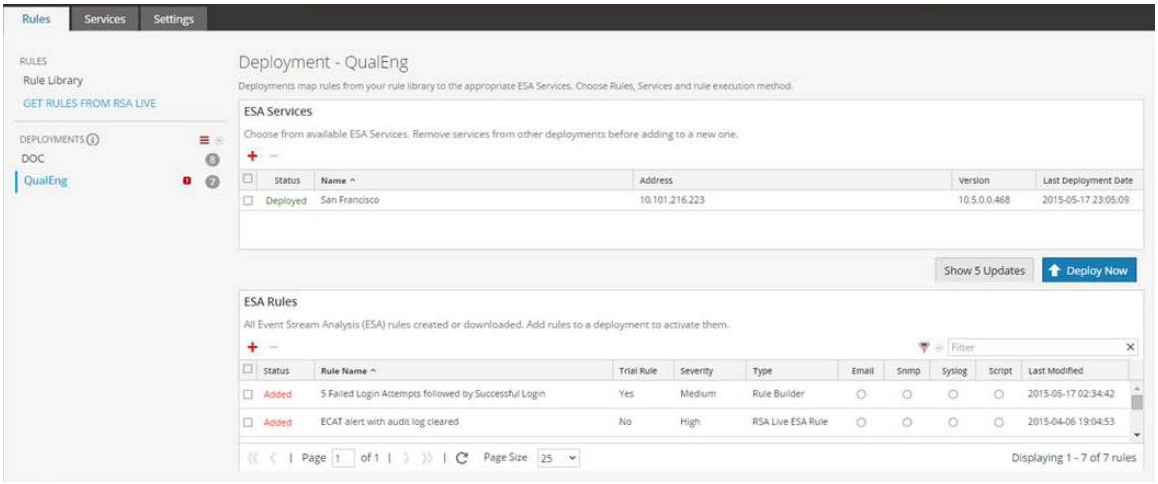
Role	I want to ...	Show me how
Content Expert	Add a deployment.	Deployment Steps
Content Expert	Manage deployments.	Additional Deployment Procedures

Related Topics

- View Stats for ESA Service

Deployment Panel

The following figure shows the Deployment panel.



ESA Services

Using the ESA Services section, you can manage each ESA service in the deployment.

In the ESA Services section, you can perform the following.

Task	Description

Task	Description
	Add an ESA service to the deployment.
	Remove the selected ESA service from the deployment.
Show Updates	Open the Updates to the Deployment dialog.
Deploy Now	Deploy current set of rules.

The following table lists the parameters of the ESA Services section.

Parameter	Description
Status	Indicates if the deployment status is **Added**, **Deployed**, **Updated**, or **Failed**.
Name	Name of the ESA service.
Address	IP address of the host where the ESA service is installed.
Version	Version of the ESA service.
Last Deployment Date	The date and time when the ESA service was last deployed.

ESA Rules

In the ESA Rules section, you manage rules in the deployment. This section lists all rules that are currently in the deployment.

In the **ESA Rules** section, you can perform the following.

Task	Description
	Open the Deploy ESA Rules dialog, where you can select a rule.
	Remove the selected ESA rules from the deployment.
	Filter the list of rules.

Task	Description
Filter	Search for a rule.

The following table lists the parameters of the ESA Rules section.

Parameter	Description
Status	Indicates the rule status: • Deployed - the rule is deployed. • Updated - the rule has been updated since the last deployment. • Added - the rule has been added since the last deployment. • Failed - the deployment failed.
Rule Name	Purpose of the ESA rule.
Trial Rule	Deployment mode to see if the rule runs efficiently.
Severity	Threat level of alert triggered by the rule.
Output	The type of the ESA rule.
Email, SNMP, Syslog, Script	Indicates which notification types are used for alerts generated by the rules.
Last Modified	The date and time when the ESA rule was last modified.

Rule Syntax Dialog

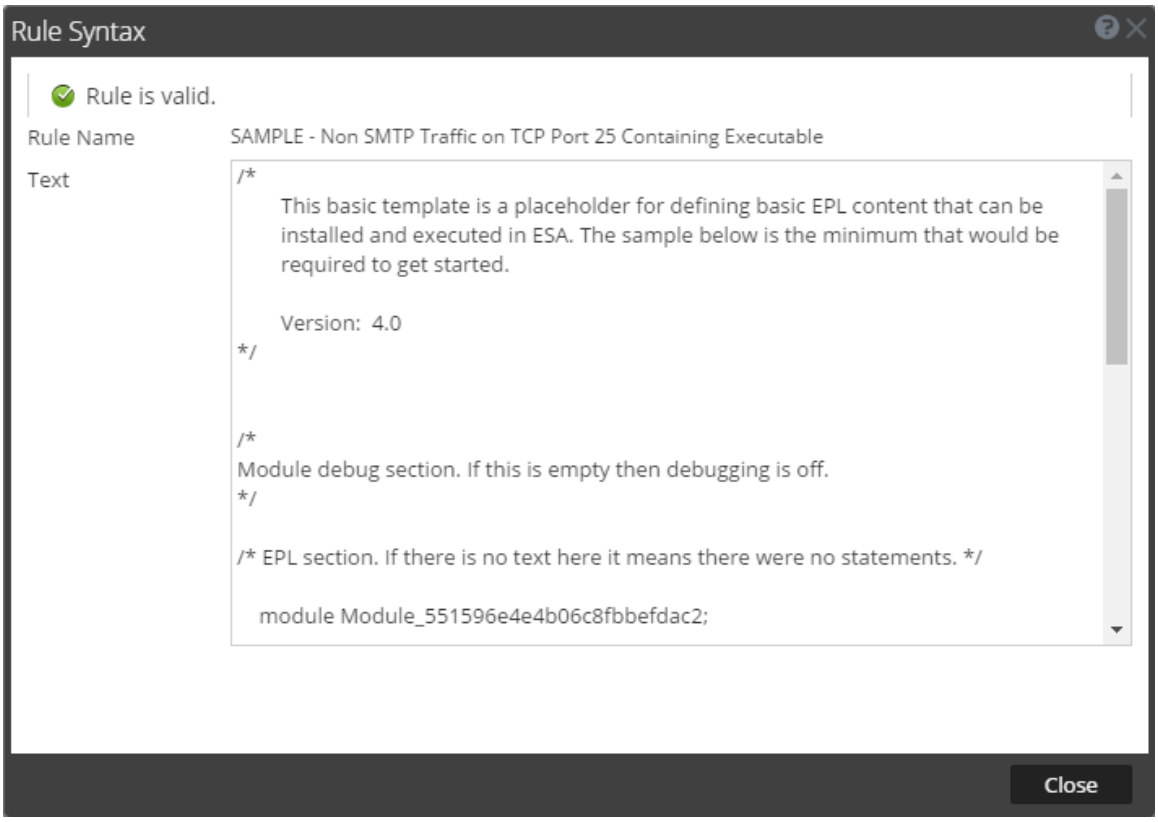
This topic describes the features of the Rule Syntax dialog. The Rule Syntax dialog displays the EPL syntax of conditions, statements, and debugging parameters, and provides a warning when the syntax is invalid.

Rule Syntax Dialog

To access this dialog:

1. Go to **CONFIGURE > ESA Rules**.
2. In the **Rule Library** view, do one of the following:
   a. Click + and select **Advanced EPL** or **Rule Builder**.
   b. Double-click an existing rule.
   c. Select an existing rule and click in the **Rule Library** toolbar.
   d. In the row of an existing rule, select > **Edit**.
   The new or existing rule is displayed in a new tab, available to edit.
3. Click **Show Syntax** at the bottom of the tab.

The following figure shows an example of the Rule Syntax dialog.



The following table describes the Rule Syntax dialog parameters.

Parameters	Description
Rule is valid or **Validation error in rule**	Indicates whether the rule syntax is valid or needs to be changed.
Rule Name	Displays the name of the rule.
Text	Displays the EPL syntax of conditions, statements, and debugging parameters if the rule is valid.

Services Tab

This topic provides an overview of the **CONFIGURE > ESA Rules > Services** tab. The Services tab provides details of the ESA services added to NetWitness Suite.

What do you want to do?

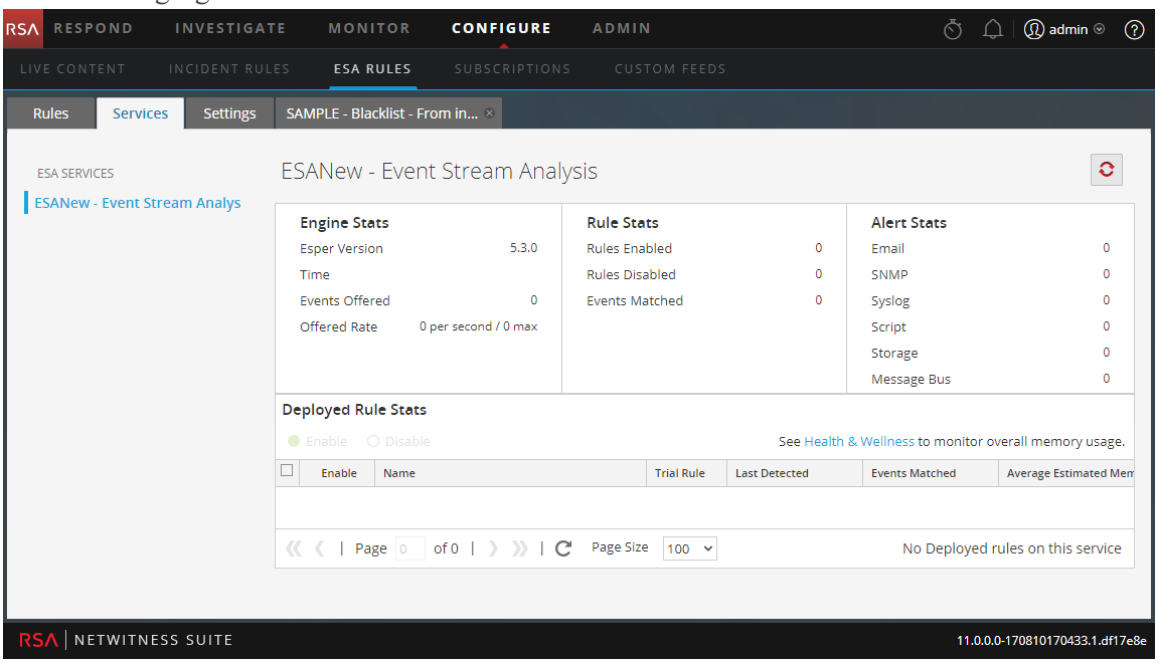
Role	I want to ...	Show me how
Content Expert	Troubleshoot Services Tab.	Troubleshoot ESA
Content Expert	View deployment Stats for an ESA Service.	View Stats for ESA Service

Related Topics

- View a Summary of Alerts

Services

The following figure shows the Services tab:



The Services tab has the following sections:

- ESA Services panel
- General Stats panel
- Deployed Rule Stats panel

ESA Services Panel

The ESA Services panel lists the name of each ESA service added to NetWitness Suite.

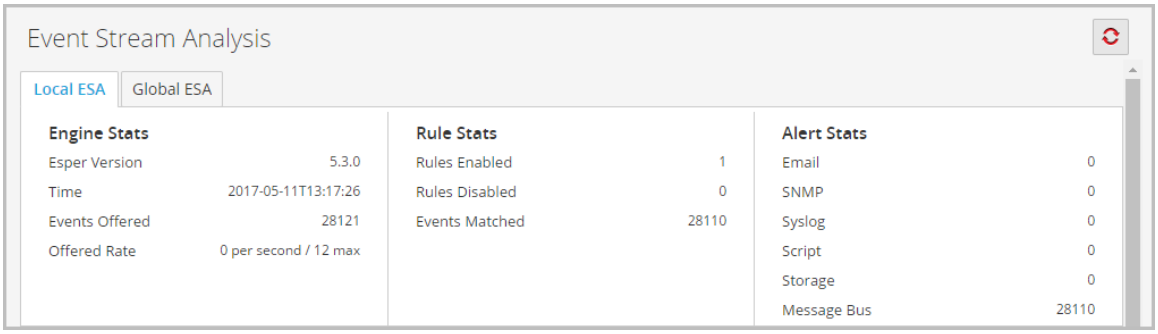
General Stats Panel

The General Stats panel provides information on the Esper engine, rules and alerts.

The General Stats panel contains the following sections:

- Engine Stats
- Rule Stats
- Alert Stats

The following figure shows the General Stats panel.



The table lists and describes the parameters in each section.

Sections	Parameter	Description
Engine Stats	Esper Version	Esper version running on the ESA service
	Time	Time when the last event was sent to Esper Engine
	Events Offered	Number of events analyzed by the ESA service since the last service start
	Offered Rate	Current events offered rate on the ESA service

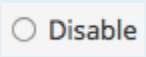
Sections	Parameter	Description
Rule Stats	Rules Enabled	Number of rules enabled.
	Rules Disabled	Number of the rules disabled
	Events Matched	Total number of events matched to all rules on the ESA service
Alert Stats	Email	Number of email notifications sent by the ESA service
	SNMP	Number of SNMP notifications sent by the ESA service
	Syslog	Number of Syslog notifications sent by the ESA service
	Script	Number of Script notifications sent by the ESA service
	Storage	Total number of alerts stored in database
	Message Bus	Total number of alerts sent to the message bus

Deployed Rule Stats Panel

The Deployed Rule Stats panel provides details on the rules that are deployed on the ESA service.

The following figure shows the Deployed Rule Stats panel.

The table lists the various parameters in the view and their description.

Parameters	Description
	Indicates the rule is enabled. Enables a rule that was disabled.
	Indicates the rule is disabled. Disables a rule that was enabled.
Health & Wellness	Displays a snapshot of memory usage when trial rules get disabled
Enable	Indicates whether the rule is enabled or disabled. Green icon indicates rule is enabled. White icon indicates rule is disabled.
Name	Name of the ESA rule.
Trial Rule	Indicates if the rule is running in trial rule mode.
Last Detected	The last time alert was triggered for the rule.
Events Matched	The total number of events that matched the rule.

Settings Tab

This topic describes the components of the **CONFIGURE > ESA Rules > Settings** tab. In the Settings tab, you can perform the following tasks:

- View a list of meta keys
- Configure a data enrichment source
- Add a connection to an external database

What do you want to do?

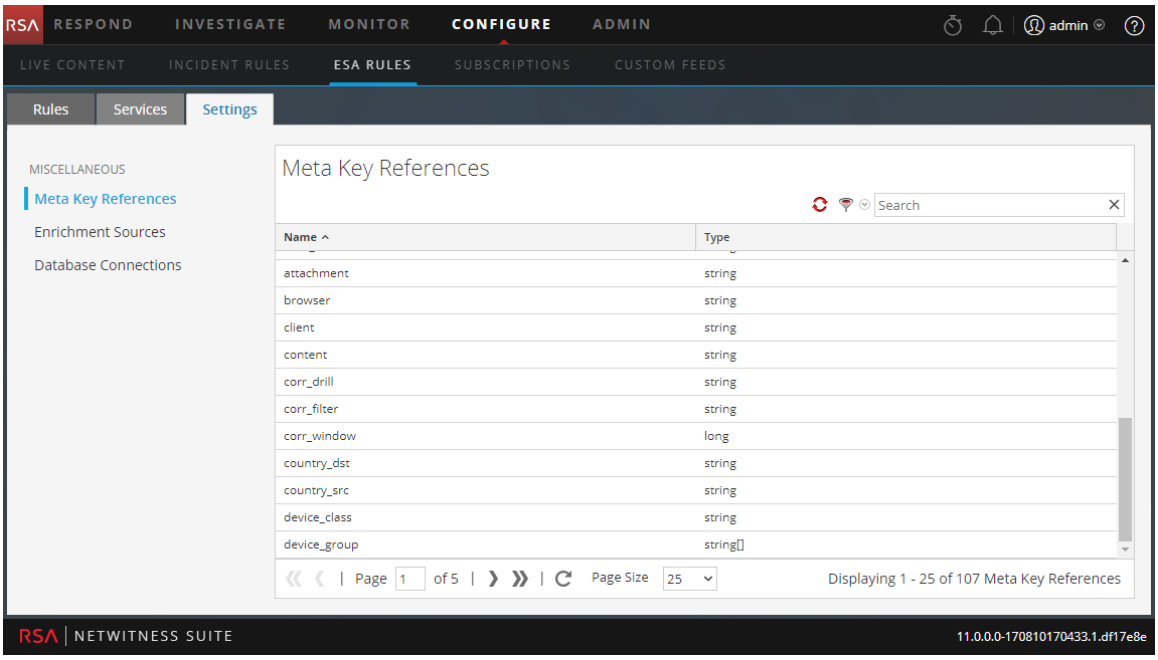
Role	I want to ...	Show me how
Content Expert	Configure a connection to an external database.	Configure a Database Connection
Content Expert	Configure a database as an enrichment source.	Enrichment Sources

Related Topics

- Add a Data Enrichment Source

Settings

The following figure shows the Meta Key References section in the Settings tab.



Meta Key References

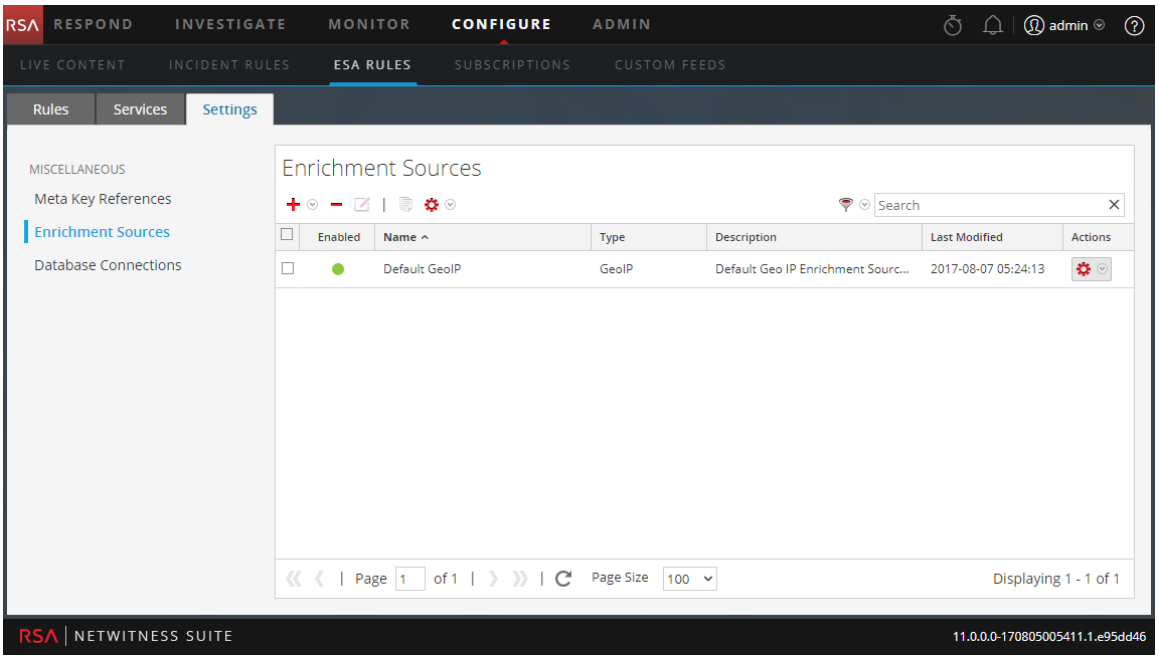
The Meta Key References section lists each meta key and the type of value the key requires.

Enrichment Sources

In the Enrichment Sources section, you can configure the following external data sources:

- GeoIP
- External Database Reference
- In-Memory Table
- Warehouse Analytics

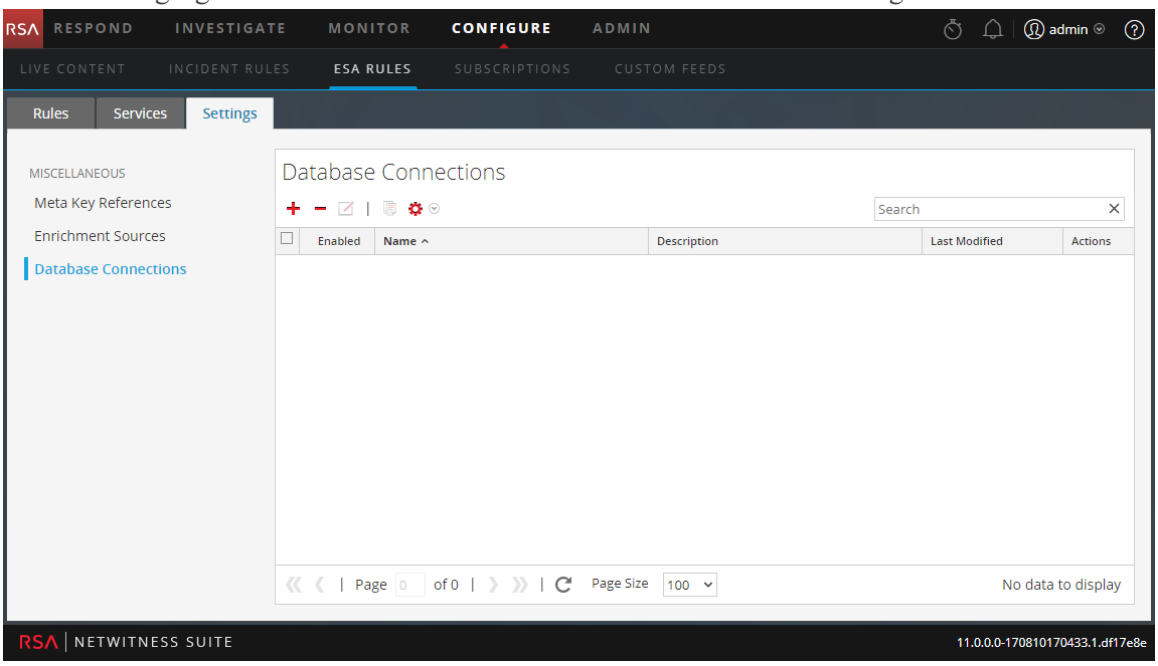
The following figure shows the Enrichment Sources section in the Settings tab.



Database Connections

In the Database Connections section, you can configure a connection to an external database so ESA can access that data.

The following figure shows the Database Connections section in the Settings tab.



In the Database Connections section you can perform the following:

- Add a Database Connection
- Delete a Database Connection
- Edit a Database Connection
- Duplicate a Database Connection
- Import a Database Connection
- Export a Database Connection

Updates to the Deployment Dialog

The Updates to the Deployment dialog displays changes to the deployment, such as adding a rule or service. Deployment updates are indicated by the update icon () next to the name of the deployment in the Rules tab options panel.

What do you want to do?

Role	I want to ...	Show me how
Content Expert	Deploy rules to run on ESA.	Deployment Steps
Content Expert	Edit or delete a deployment.	Edit or Delete a Deployment
Content Expert	View deployment updates.	Show Updates to a Deployment

Related Topics

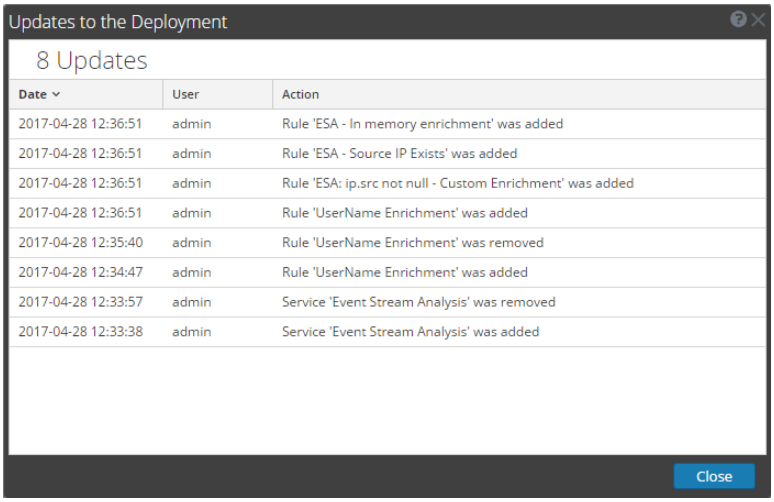
- Delete ESA Service in a Deployment
- Edit or Delete Rule in a Deployment

Deployment Dialog

To access this dialog:

1. Go to **CONFIGURE > ESA Rules**.
   The Rules tab opens by default.
2. In the options panel, under the **Deployments** section, select or add a deployment.
3. In the **Deployment** panel, click **Show Updates**.
   The Updates to the Deployment dialog is displayed.

The following figure is an example of this dialog.



The Updates to the Deployment dialog displays the number of updates at the top of the dialog. The following table describes the parameters of this dialog.

Parameters	Description
Date	Displays the day and time of the update.
User	Displays the user who made the update.
Action	Describes the update.